Oracle® Application Server

Adapter for PeopleSoft User's Guide

10*g* Release 2 (10.1.2)

B14060-03

April 2006

Oracle Application Server Adapter for PeopleSoft User's Guide, 10*g* Release 2 (10.1.2)

B14060-03

Primary Author: Ivanka J. Stoycheva

Contributing Authors: Vishal Saxena, Sunil Gopal, Sheela Vasudevan

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software—Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

3 OC4J Deployment and Integration

4 Integration with BPEL Process Manager

5 BPEL Process Manager Integration Examples

6 InterConnect Integration Examples

7 Troubleshooting and Error Messages

8 Advanced User Tools

A Generating Component Interface APIs

B Configuring the PeopleSoft Message Router

C Using Component Interfaces

D Using PeopleSoft Integration Broker

Glossary

Index

Preface

This Preface contains these topics:

- Audience
- Documentation Accessibility
- Related Documents
- Conventions
- Help Us to Serve You Better

Audience

Oracle Application Server Adapter for PeopleSoft User's Guide is intended for those who perform the following tasks:

- Install applications
- Maintain applications

To use this document, you need to know how to install and configure Oracle BPEL Process Manager.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

`http://www.oracle.com/accessibility/`

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Related Documents

For more information, refer to these Oracle resources:

- *Oracle Application Server Adapter Concepts*
- *Oracle Application Server Adapters Installation Guide*

Printed documentation is available for sale in the Oracle Store at

`http://oraclestore.oracle.com/`

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

`http://www.oracle.com/technology/membership/`

If you already have a user name and password for OTN, then you can go directly to the documentation section of the OTN Web site at

`http://www.oracle.com/technology/documentation/`

Conventions

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- Conventions in Text
- Conventions in Code Examples
- Conventions for Windows Operating Systems

Conventions in Text

We use the following conventions in text to help you more quickly identify special terms. The table also provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an **index-organized table**.
Italics	Italic typeface indicates book titles or emphasis.	*Oracle Database 10g Concepts* Ensure that the recovery catalog and target database do *not* reside on the same disk.

Convention	Meaning	Example
UPPERCASE monospace (fixed-width) font	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, Recovery Manager keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, user names, and roles.	You can specify this clause only for a NUMBER column. You can back up the database by using the BACKUP command. Query the TABLE_NAME column in the USER_TABLES data dictionary view. Use the DBMS_STATS.GENERATE_STATS procedure.
lowercase monospace (fixed-width) font	Lowercase monospace typeface indicates executable programs, filenames, directory names, and sample user-supplied elements. *Note:* Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter sqlplus to start SQL*Plus. The password is specified in the orapwd file. Back up the datafiles and control files in the /disk1/oracle/dbs directory. The department_id, department_name, and location_id columns are in the hr.departments table. Connect as oe user. The JRepUtil class implements these methods.
lowercase italic monospace (fixed-width) font	Lowercase italic monospace font represents placeholders or variables.	You can specify the *parallel_clause*. Run *old_release*.SQL where *old_release* refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
[ ]	Anything enclosed in brackets is optional.	DECIMAL (*digits* [, *precision*])
{ }	Braces are used for grouping items.	{ENABLE | DISABLE}
|	A vertical bar represents a choice of two options.	{ENABLE | DISABLE} [COMPRESS | NOCOMPRESS]

. . .	Ellipsis points mean repetition in syntax descriptions.
	In addition, ellipsis points can mean an omission in code examples or text.
		CREATE TABLE ... AS *subquery*;
		SELECT *col1*, *col2*, ... , *coln* FROM employees;

Other symbols	You must use symbols other than brackets ([ ]), braces ({ }), vertical bars (|), and ellipsis points (...) exactly as shown.	acctbal NUMBER(11,2);
		acct CONSTANT NUMBER(4) := 3;

Italics	Italicized text indicates placeholders or variables for which you must provide particular values.	CONNECT SYSTEM/*system_password* DB_NAME = *database_name*

Convention	Meaning	Example

`UPPERCASE`	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. Because these terms are not case sensitive, you can use them in either UPPERCASE or lowercase.	`SELECT last_name, employee_id FROM employees;`
		`SELECT * FROM USER_TABLES;`
		`DROP TABLE hr.employees;`

`lowercase`	Lowercase typeface indicates user-defined programmatic elements, such as names of tables, columns, or files.
	Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.
		`SELECT last_name, employee_id FROM employees;`
		`sqlplus hr/hr`
		`CREATE USER mjones IDENTIFIED BY ty3MU9;`

Conventions for Windows Operating Systems

The following table describes conventions for Windows operating systems and provides examples of their use.

Convention	Meaning	Example
Click **Start**, and then choose the ***menu item***	How to start a program.	To start the Database Configuration Assistant, click **Start**, and choose **Programs**. In the Programs menu, choose **Oracle** - ***HOME_NAME*** and then click **Configuration and Migration Tools**. Choose **Database Configuration Assistant**.
File and directory names	File and directory names are not case sensitive. The following special characters are not allowed: left angle bracket (<), right angle bracket (>), colon (:), double quotation marks ("), slash (/), pipe (|), and dash (-). The special character backslash (\) is treated as an element separator, even when it appears in quotes. If the filename begins with \\, then Windows assumes it uses the Universal Naming Convention.	c:\winnt"\"system32 is the same as C:\WINNT\SYSTEM32
`C:\>`	Represents the Windows command prompt of the current hard disk drive. The escape character in a command prompt is the caret (^). Your prompt reflects the subdirectory in which you are working. Referred to as the *command prompt* in this manual.	`C:\oracle\oradata>`
Special characters	The backslash (\) special character is sometimes required as an escape character for the double quotation mark (") special character at the Windows command prompt. Parentheses and the single quotation mark (') do not require an escape character. Refer to your Windows operating system documentation for more information on escape and special characters.	`C:\>exp HR/HR TABLES=employees QUERY=\"WHERE job_id='SA_REP' and salary<8000\"`

Convention	Meaning	Example
HOME_NAME	Represents the Oracle home name. The home name can be up to 16 alphanumeric characters. The only special character allowed in the home name is the underscore.	`C:\> net start Oracle`*`HOME_NAME`*`TNSListener`
ORACLE_HOME and *ORACLE_BASE*	In releases prior to Oracle8*i* release 8.1.3, when you installed Oracle components, all subdirectories were located under a top level *ORACLE_HOME* directory. This release complies with Optimal Flexible Architecture (OFA) guidelines. All subdirectories are not under a top level *ORACLE_HOME* directory. There is a top level directory called *ORACLE_BASE* that by default is `C:\oracle\product\10.1.0`. If you install the latest Oracle release on a computer with no other Oracle software installed, then the default setting for the first Oracle home directory is `C:\oracle\product\10.1.0\db_`*n*, where *n* is the latest Oracle home number. The Oracle home directory is located directly under *ORACLE_BASE*. All directory path examples in this guide follow OFA conventions. Refer to *Oracle Database Installation Guide for Windows* for additional information about OFA compliances and for information about installing Oracle products in non-OFA compliant directories.	Change to the *ORACLE_BASE*`\`*ORACLE_HOME*`\rdbms\admin` directory.

Help Us to Serve You Better

To help our consultants answer your questions effectively, please be prepared to provide specifications and sample files and to answer questions about errors and problems.

The following list includes the specifications our consultants require.

- **Platform**:
- **Operating System**:
- **Operating System Version**:
- **Product List**:
- **Adapters**:
- **Adapter Deployment**:

  For example, *J2CA* or *Business Services Engine (BSE)*
- **Container Version**:

The following table lists components. Specify the version in the column provided.

Component	Version
Adapter	
EIS (DBMS/APP)	
HOTFIX/Service Pack	

In the following table, specify the JVM version and vendor.

JVM Version	Vendor

The following table lists additional questions to help us serve you better.

Request/Question	Error/Problem Details or Information
Provide usage scenarios or summarize the application that produces the problem.	
Has this happened previously?	
Can you reproduce this problem consistently?	
Any **change in the application environment**: software configuration, EIS/database configuration, application, and so on?	
Under what circumstance does the problem *not* occur?	
Describe the **steps** to reproduce the problem.	
Describe the **problem**.	
Specify the **error** message(s).	

The following is a list of error or problem files that might be applicable.

- XML schema
- XML instances
- Other input documents (transformation)
- Error screen shots
- Error output files
- Trace and log files
- Log transaction

1 Introduction

Oracle Application Server connects to a PeopleSoft system through Oracle Application Server Adapter for PeopleSoft (OracleAS Adapter for PeopleSoft). OracleAS Adapter for PeopleSoft provides connectivity and carries out interactions on a PeopleSoft system. This chapter discusses the following topics:

- Adapter Features
- PeopleSoft Concepts
- Integration with PeopleSoft
- Adapter Architecture
- BSE Versus OracleAS Adapter J2CA Deployment

Adapter Features

OracleAS Adapter for PeopleSoft provides a means to exchange real-time business data between PeopleSoft systems and other applications, databases, and external business partner systems. The adapter enables inbound and outbound processing with PeopleSoft. OracleAS Adapter for PeopleSoft can be deployed as a J2EE Connector Architecture (J2CA) version 1.0 resource adapter. This deployment is referred to as OracleAS Adapter J2CA. It can also be deployed as a Web services servlet and is referred to as Oracle Application Server Adapter Business Services Engine (BSE).

OracleAS Adapter for PeopleSoft uses XML messages to enable non-PeopleSoft applications to communicate and exchange transactions with PeopleSoft using services and events. The roles of services and events are outlined as follows:

- Services: Enable applications to initiate a PeopleSoft business event.
- Events: Enable applications to access PeopleSoft data only when a PeopleSoft business event occurs.

To support event functionality, the following two features are implemented:

- Port: A **port** associates a particular business object exposed by an adapter with a particular disposition. A disposition defines the protocol and location of the event data. The port defines the end point of the event consumption.

  The port is the Oracle adapter component that pushes the event received from the enterprise information system (EIS) to the adapter client. The port supported in this release is Remote Method Invocation (RMI).

Note: You are not required to create or configure ports for use with BPEL Process Manager. However, in this release you can associate an event schema to a port under a J2CA configuration. See "J2CA Filtering Port Option for Integration with BPEL Process Manager" on page 2-20 for details on using ports under a J2CA configuration.

- Channel: A **channel** represents configured connections to particular instances of back-end or other types of systems. A channel binds one or more event ports to a particular **listener** managed by an adapter.

  The channel is the adapter component that receives events in real time from the enterprise information system (EIS) application. The channel component can be a File reader, an HTTP listener, a TCP/IP listener, or an FTP listener. A channel is always EIS specific. The adapter supports multiple channels for a particular EIS. This enables the user to choose the optimal channel component based on deployment requirements.

OracleAS Adapter for PeopleSoft provides:

- Synchronous and asynchronous, bidirectional message interactions for PeopleSoft component interfaces and PeopleSoft Application Messaging Manager.
- Oracle Application Server Adapter Application Explorer (Application Explorer), a GUI tool that explores PeopleSoft metadata and builds XML schemas or Web services.
- XML schemas for the J2CA 1.0 resource adapter.
- Web services for BSE.

The adapter connects to the PeopleSoft Application Server by accessing APIs for the component interfaces that correspond to its supported business objects. Every component interface contains data and business logic for the business component, thus alleviating a requirement for the adapter to duplicate the processes defined within the business component.

See Also: *Oracle Application Server Adapter Concepts*

PeopleSoft Concepts

PeopleSoft enables integration with other applications and systems through its component interface framework and its Integration Broker (in release 8.4) or Application Messaging (in release 8.1) facility. OracleAS Adapter for PeopleSoft uses the PeopleSoft framework and leverages various integration access methods to provide the greatest amount of flexibility and functionality. Integration access methods supported by OracleAS Adapter for PeopleSoft include:

- PeopleSoft Java API using component interfaces
- PeopleSoft XML using Application Messaging

PeopleSoft Component Interface

In the PeopleSoft environment, a component interface is a container for distributing PeopleSoft application data among PeopleSoft logical systems and for exchanging PeopleSoft application data with non-PeopleSoft systems.

The component interface is based on an existing business process within PeopleSoft. An example is a purchase order entry, which can be a PeopleSoft-delivered process or a user-developed process. The component interface also inherits its methods (Add, Update, and so on) and its business logic from the underlying business process.

PeopleSoft delivers generic component interfaces with each of its applications. These are called Enterprise Integration Points (EIP). Customers can also develop their own custom component interfaces, or they can modify EIP as required. OracleAS Adapter for PeopleSoft supports both types of component interfaces.

PeopleSoft Application Messaging Manager

PeopleSoft Application Messaging Manager facilitates the integration of PeopleSoft XML with PeopleSoft. OracleAS Adapter for PeopleSoft provides a handler that must be configured within the PeopleSoft application gateway using TCP/IP transport services.

Integration with PeopleSoft

OracleAS Adapter for PeopleSoft enables you to:

- Access a PeopleSoft component using a component interface. Component methods are also referred to as services.
- Receive messages from a PeopleSoft application. Messages are also referred to as events.

When you access a PeopleSoft component from another application, you work with:

- Component interfaces. If a component interface does not exist, create, secure, and test one. If the component interface exists but you modified it, secure and test it. For more information, see your PeopleSoft documentation. Alternatively, you can secure and test the component interface and create the component interface API after you generate schemas or Web services.
- Component interface APIs. Create an API for the component interface.
- Schemas and Web services. Create schemas or Web services for the component methods.

See Chapter 2, "Configuring OracleAS Adapter for PeopleSoft" for more information.

To receive a message from PeopleSoft, you work with:

- The Integration environment. Configure and test your PeopleSoft Integration Broker (release 8.4) or Application Messaging environment (release 8.1). To ensure that the environment is properly configured, see your PeopleSoft documentation.
- Message routing. Configure TCP/IP Target Connector (release 8.4), HTTP Target Connector (release 8.4), or TCP/IP Handler (release 8.1).

See Also:

- *Oracle Application Server Adapter Concepts*
- *Oracle Application Server Adapters Installation Guide*

Adapter Architecture

OracleAS Adapter for PeopleSoft works with Application Explorer in conjunction with one of the following components:

- Oracle Application Server Adapter Business Services Engine (BSE)

- Enterprise Connector for J2EE Connector Architecture (J2CA)

Application Explorer (used to configure PeopleSoft connections and create Web services and events) can be configured to work in a Web services environment in conjunction with BSE. When working in a J2CA environment, the connector uses the Common Client Interface (CCI) to provide integration services using adapters instead of Web services.

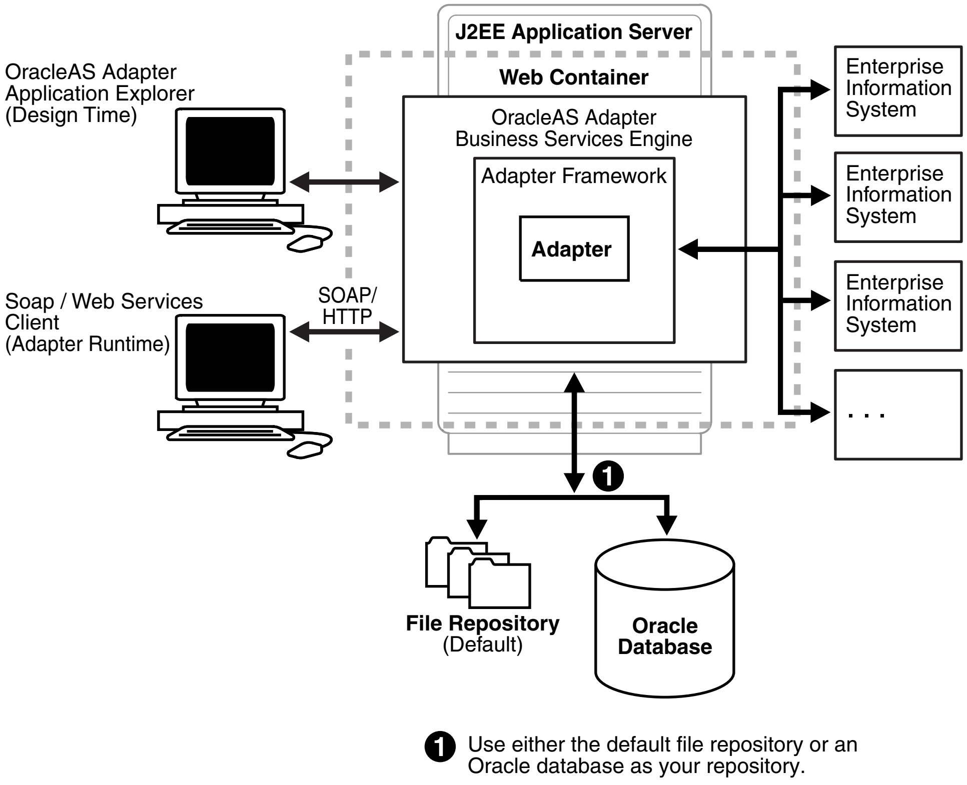
Oracle Application Server Adapter Business Services Engine (BSE) Architecture

Figure 1–1 shows the generic architecture for the Oracle Web service adapter for packaged applications. The adapter works in conjunction with BSE, as deployed to a Web container in a J2EE application server.

Application Explorer, a design-time tool deployed along with BSE, is used to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. Metadata created while you perform these operations are stored in the repository by BSE.

BSE uses SOAP as a protocol for receiving requests from clients, interacting with the EIS, and sending responses from the EIS back to clients.

Figure 1–1 Oracle Application Server Adapter Business Services Engine (BSE) Generic Architecture


J2EE Application Server
Web Container
OracleAS Adapter Application Explorer (Design Time)
OracleAS Adapter Business Services Engine
Adapter Framework
Adapter
Soap / Web Services Client (Adapter Runtime)
SOAP/ HTTP
Enterprise Information System
Enterprise Information System
Enterprise Information System
. . .
1
File Repository (Default)
Oracle Database
1 Use either the default file repository or an Oracle database as your repository.

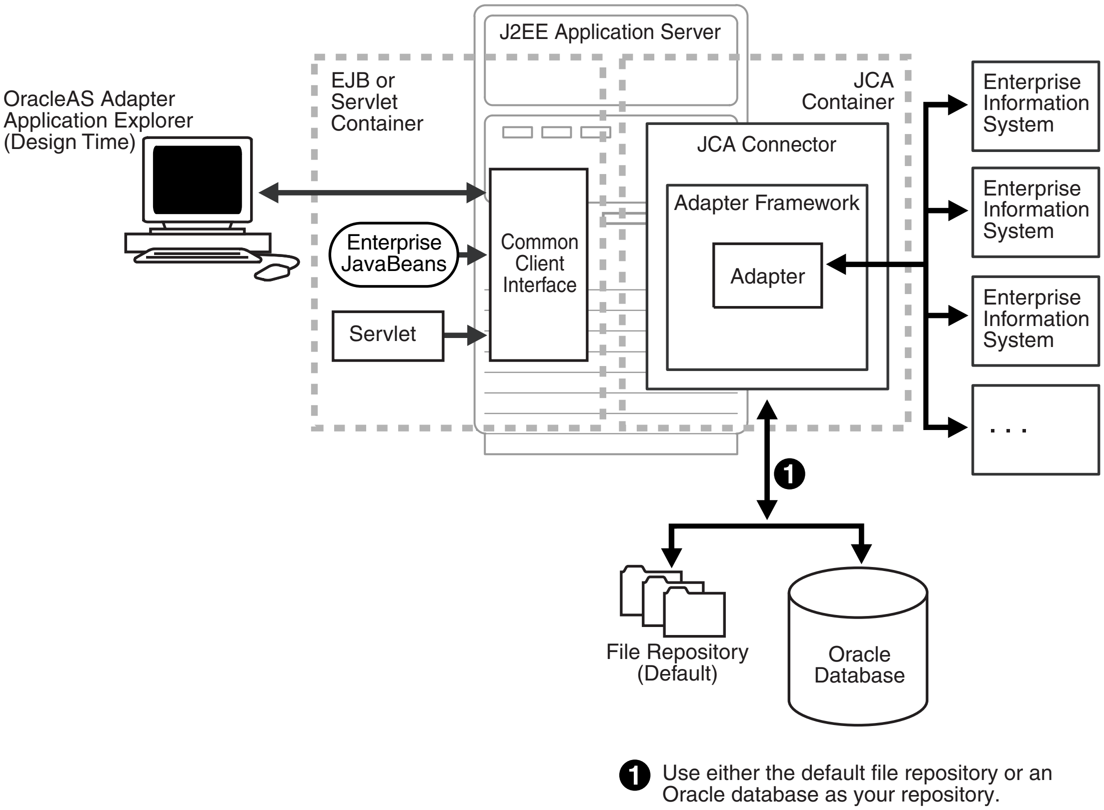
Note: Do not use a file repository for BSE in production environments.

Oracle Application Server Adapter J2CA Architecture

Figure 1–2 shows the generic architecture for OracleAS Adapter J2CA for packaged applications. OracleAS Adapter J2CA is deployed to a standard J2CA Container and serves as a host container to the adapters. The connector is configured with a repository. The repository can be a file system or an Oracle database. It is deployed as a RAR file and has an associated deployment descriptor called `ra.xml`. You can create multiple connector factories by editing the OC4J deployment descriptor `oc4j-ra.xml`. See Chapter 3, "OC4J Deployment and Integration" for more information on OC4J deployment.

Application Explorer, a design tool that works in conjunction with the connector, is used to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. Metadata created while you perform these operations are stored in the repository by the connector.

Figure 1–2 Oracle Application Server Adapter J2CA Generic Architecture



1 Use either the default file repository or an Oracle database as your repository.

BSE Versus OracleAS Adapter J2CA Deployment

If you are using OracleAS Adapter for PeopleSoft with BPEL Process Manager, please note that:

- Only OracleAS Adapter J2CA deployment supports inbound integration (event notification) with BPEL Process Manager.
- Both OracleAS Adapter J2CA and BSE deployments support outbound integration (request-response service) with BPEL Process Manager.

The following three factors explain the differences between deploying the BSE and OracleAS Adapter J2CA. Understanding the factors can help in selecting a deployment option.

1. BSE is the preferred deployment option because it:
   - Can be deployed in a separate instance of Oracle Application Server.
   - Provides better distribution of load.
   - Provides better isolation from any errors from third party libraries.
   - Provides better capability to isolate issues for debugging purposes.
   - Conforms more closely to the Service Oriented Architecture (SOA) model for building applications.
2. OracleAS Adapter J2CA provides slightly better performance.

   OracleAS Adapter J2CA does provide slightly better performance than BSE. However, the difference decreases as the transaction rate increases.
3. OracleAS Adapter J2CA and the BSE option both provide identity propagation at runtime.

   The BSE option provides the capability to pass identity using the SOAP header. For OracleAS Adapter J2CA, user name and password can be passed using the connection specification of the CCI.

2

Configuring OracleAS Adapter for PeopleSoft

OracleAS Adapter Application Explorer (Application Explorer) enables the processing of Component Interfaces and Messages.

External applications that access PeopleSoft through the adapter use either XML schemas or Web services to pass data between the external application and the adapter. You can use Application Explorer to create the required XML schemas and Web services.

This chapter discusses the following topics:

Starting Application Explorer

To start Application Explorer:

1. Ensure the server is started where Application Explorer is deployed.
2. On Windows, select **Start** ,**Programs**, ***OracleAS_home*** **Adapters**, **Application Explorer**.

   On Windows, `iaexplorer.bat` is found under `*OracleAS_home*\adapters\application\tools`, where `OracleAS_home` is the directory where Oracle Application Server is installed.

   On UNIX, load the iwae script, `iwae.sh`, found under `*OracleAS_home*/adapters/application/tools`, where `OracleAS_home` is the directory where Oracle Application Server is installed.

Application Explorer starts. You are ready to define new targets to your PeopleSoft system.

Configuring Settings for BSE or J2CA

Before a repository project can be created, you must configure BSE. You need not configure the OracleAS Adapter J2CA because the `ra.xml` file is configured automatically during installation.

Configuring BSE

After BSE is deployed to Oracle Application Server, you can configure it through the BSE configuration page. This configuration is required only when pointing to BSE using a database repository.

> **Note:** Do not use a file repository for BSE in production environments.

To configure BSE:

1. Display the following page in your browser:

   ```
   http://hostname:port/ibse
   ```

   Where `hostname` is the host name of Oracle Application Server and `port` is the HTTP port for Oracle Application Server.

   For example,

   ```
   http://localhost:7777/ibse
   ```

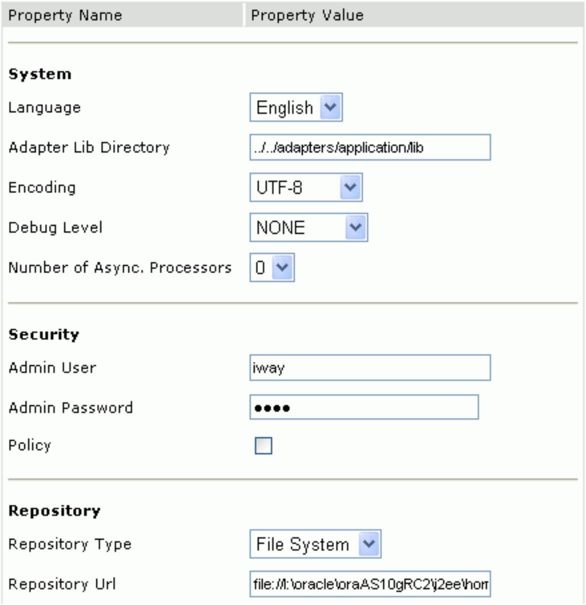
   > **Note:** If you are accessing this page for the first time, it may take longer to load.

2. When prompted, log on.

   When first installed, the user ID and password are:

   - User name: iway
   - Password: iway

     The BSE configuration page is displayed.



Property Name	Property Value
System	
Language	English
Adapter Lib Directory	../../adapters/application/lib
Encoding	UTF-8
Debug Level	NONE
Number of Async. Processors	0
Security	
Admin User	iway
Admin Password	●●●●
Policy	□
Repository	
Repository Type	File System
Repository Url	file:///I:\oracle\oraAS10gRC2\j2ee\hom

3. Ensure that the Adapter Lib Directory parameter specifies the path to the lib directory, for example:

   ```
   OracleAS_home\adapters\application\lib
   ```

   After you specify the path, adapters in the lib directory are available to BSE.

4. For security purposes, enter a new password in the **Admin Password** field.

   > **Note:** The Repository URL field specifies where the file system repository is located. To use a database repository, you must enter the repository connection information. For the initial verification, use a file system repository. See "Configuring an Oracle Repository" on page 2-6 for information on switching to a database repository.

5. Click **Save**.

Configuring BSE System Settings

To configure BSE system settings:

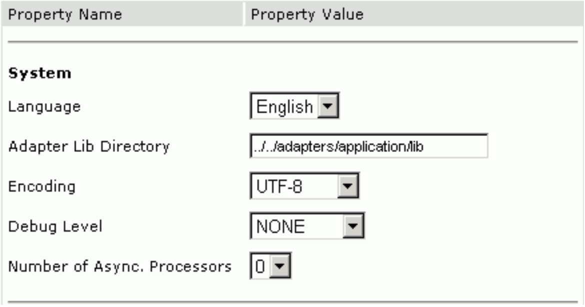
1. Display the **BSE configuration** page by entering the following URL in a browser:

   ```
   http://hostname:port/ibse/IBSEConfig
   ```

   Where `hostname` is the machine where BSE is installed and `port` is the port number on which BSE is listening.

   > **Note:** The server to which BSE is deployed must be running.

The BSE settings pane is displayed, as shown in the following figure.



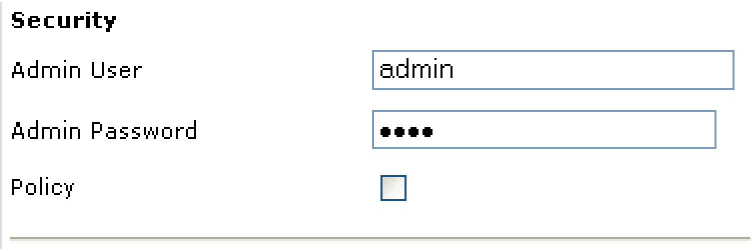
Property Name	Property Value
System	
Language	English
Adapter Lib Directory	../../adapters/application/lib
Encoding	UTF-8
Debug Level	NONE
Number of Async. Processors	0

2. Configure the system settings.

   The following table lists the parameters with descriptions of the information to provide.

Parameter	Description
Language	Specify the required language.
Adapter Lib Directory	Enter the full path to the directory where the adapter jar files reside.
Encoding	Specify the default encoding from one of the following options: ■ UTF-8 ■ EBCDIC-CP-US ■ ISO-88859-1 ■ Shift JIS ■ UNICODE
Debug Level	Specify the debug level from one of the following options: ■ None ■ Fatal ■ Error ■ Warning ■ Info ■ Debug
Number of Async. Processors	Select the number of asynchronous processors.

3. Configure the security settings.



Security	
Admin User	admin
Admin Password	●●●●
Policy	□

   The following table lists the parameters with descriptions of the information to provide.

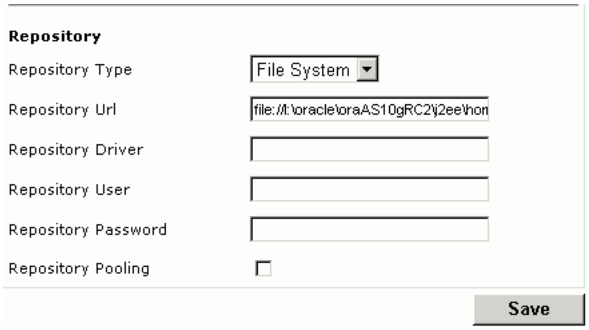
Parameter	Description
Admin User	Provide a BSE administrator ID.
Admin Password	Enter the password associated with the BSE administrator ID.
Policy	Select the check box to enable policy security.

4. Configure the repository settings.

   BSE requires a repository to store transactions and metadata required for the delivery of Web services.

   See "Configuring a File System Repository" on page 2-5 and "Configuring an Oracle Repository" on page 2-6 for more information.

   The following image shows all fields and check boxes for the Repository pane:



Repository

Repository Type: File System

Repository Url: file://I:\oracle\oraAS10gRC2\j2ee\hon

Repository Driver:

Repository User:

Repository Password:

Repository Pooling: □

   The following table lists the parameters with descriptions of the information to provide.

Parameter	Description
Repository Type	Select one of the following repositories from the list: - Oracle - File (Do not use a file repository for BSE in production environments.)
Repository URL	Enter the URL to use when opening a connection to the database.
Repository Driver	Provide the driver class to use when opening a connection to the database (optional).
Repository User	Enter the user ID to use when opening a connection to the database.
Repository Password	Enter the password associated with the user ID.
Repository Pooling	Select the check box to enable pooling.

5. Click **Save**.

Configuring a File System Repository

If you do not have access to a database for the repository, you can store repository information in an XML file on your local machine. However, a file system repository is less secure and efficient than a database repository. When BSE is first installed, it is automatically configured to use a file system repository.

Note: Do not use a file repository for BSE in production environments.

The default location for the repository on Windows is:

```
OracleAS_home\j2ee\OC4J_CONTAINER\applications\ws-app-adapter
\ibse\ibserepo.xml
```

On other platforms, use the corresponding location.

If you are using a file system repository, you are not required to configure any additional BSE components.

Configuring an Oracle Repository

To configure an Oracle repository:

1. Contact your database administrator to obtain an Oracle user ID and password to create the BSE repository.

   This user ID should have rights to create and modify tables as well as the ability to create and execute stored procedures.

2. Open a command prompt and navigate to the setup directory. Its default location on Windows is:

   ```
   OracleAS_home\adapters\application\etc
   ```

   For other platforms, see the corresponding location.

   This directory contains SQL to create the repository tables in the following file:

   ```
   iwse.ora
   ```

   Note: If Oracle is not on the same machine as the Oracle Application Server, copy the iwse.ora file to the Oracle machine. Then, from a command prompt on the Oracle machine, navigate to the directory containing the iwse.ora file.

3. Enter the following command:

   ```
   sqlplus userid/password @database @ iwse.ora
   ```

Configuring J2CA

During the J2CA deployment of OracleAS Adapter for PeopleSoft, OC4J generates a deployment descriptor called `oc4j-ra.xml`. This descriptor provides OC4J-specific deployment information for resource adapters. See Chapter 3, "OC4J Deployment and Integration" for more information on J2CA deployment and configuration.

No configuration changes are necessary if you are using the default file based repository with J2CA deployment.

Configuring a Database Repository for J2CA

To configure a database repository for J2CA:

1. Execute the `iwse.ora` SQL statement on the machine where the database is installed.

2. Copy the `jcatransport.properties` file to the following directory:

```
OracleAS_home\adapters\application\config\jca_sample
```

3. Uncomment the following fields and enter details for them in the `jcatransport.properties` file. For example:

```
iwafjca.repo.url=jdbc:oracle:thin:@90.0.0.51:1521:orcl
iwafjca.repo.user=scott
iwafjca.repo.password=scott1
```

4. Alter the JDBC driver path in Application Explorer's lcp. For example:

```
lcp=..\lib\orabpel-adapters.jar;C:\jdev\jdbc\lib\classes12.jar;C:\jdev\jdbc\lib
\nls_charset12.jar;%lcp%
to
lcp=..\lib\orabpel-adapters.jar;..\..\..\jdbc\lib\classes12.jar;..\..\..\jdbc\l
ib\nls_charset12.jar;%lcp%
```

Creating a Repository Configuration

Before you use Application Explorer with OracleAS Adapter for PeopleSoft, you must create a repository configuration. You can create two kinds of repository configurations, Web services and J2CA, depending on the container to which the adapter is deployed.

During design time, the repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at runtime.

A default J2CA repository is created for the default ManagedConnectionFactory. The name of this configuration is `jca_sample`.

Web services and BSE refer to the same type of deployment. See "Adapter Features" on page 1-1 for more information.

Creating a Configuration for BSE

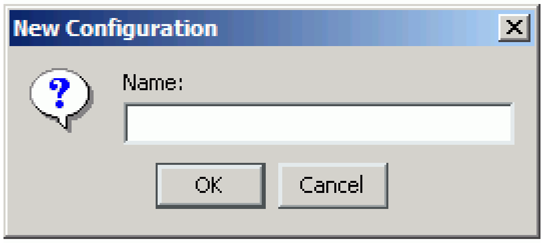
To create a repository configuration for BSE using Application Explorer, you must first define a new configuration.

Defining a New Configuration for BSE

To define a new configuration for BSE:

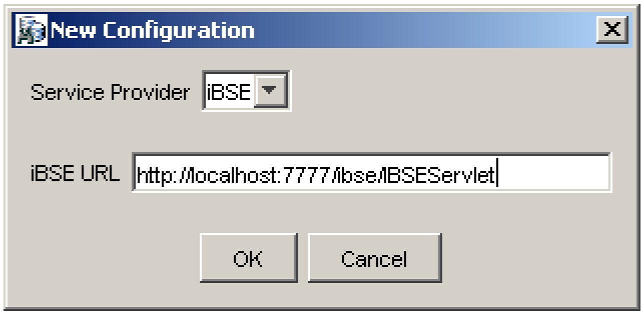
1. Right-click **Configurations** and select **New**.

   The New Configuration dialog box is displayed.



2. Enter a name for the new configuration (for example, myConfig) and click **OK**.

The New Configuration dialog box is displayed.



Service Provider: iBSE

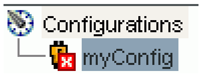
iBSE URL: http://localhost:7777/ibse/IBSEServlet

3. From the **Service Provider** list, select **iBSE**.
4. In the **BSE URL** field, accept the default URL or replace it with a different URL with the following format:

   ```
   http://hostname:port/ibse/IBSEServlet
   ```

   Where `hostname` is the machine where your application server resides and `port` is the port number where the application server is listening.
5. Click **OK**.

   A node representing the new configuration appears beneath the root Configurations node.



Creating a Configuration for J2CA

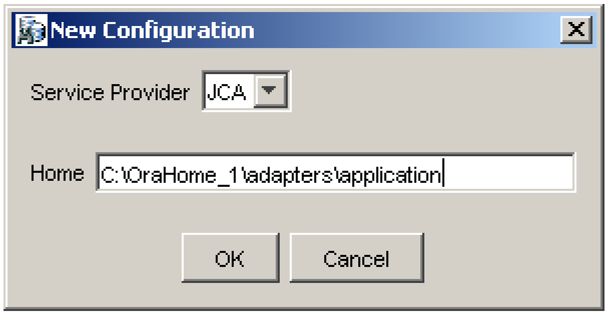
To create a configuration for OracleAS Adapter J2CA using Application Explorer, you must first define a new configuration.

Defining a New Configuration for J2CA

To define a new configuration for J2CA:

1. Right-click **Configurations** and select **New**.

   The New Configuration dialog box is displayed.
2. Enter a name for the new configuration (for example, myConfig) and click **OK**.



Service Provider: JCA

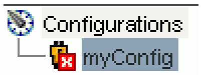
Home: C:\OraHome_1\adapters\application

3. From the **Service Provider** list, select **JCA**.
4. In the **Home** field, enter a path to your J2CA configuration directory where the repository, schemas, and other information is stored, for example:

OracleAS_home\adapters\application

5. Click **OK**.

   A node representing the new configuration appears beneath the root Configurations node.



The OracleAS Adapter J2CA configuration file is stored in *OracleAS_home*\adapters\application\config\configuration_name

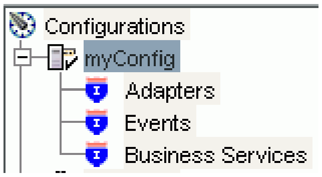
Where configuration_name is the name of the configuration you created; for example, myConfig.

Connecting to a BSE or J2CA Configuration

To connect to a new configuration:

1. Right-click the configuration to which you want to connect, for example, myConfig.
2. Select **Connect**.

Nodes appear for Adapters, Events, and Business Services (also known as Web services). The Business Services node is only available for BSE configurations. If you are connected to a J2CA configuration, you will not see the Business Services node. The following is an example of a BSE configuration named myConfig:



- Use the **Adapters** folder to create inbound interaction with PeopleSoft. For example, you use the PeopleSoft node in the Adapters folder to configure a service that updates PeopleSoft.
- Use the **Events** folder to configure listeners that listen for events in PeopleSoft.
- Use the **Business Services** folder (available for BSE configurations only) to test Web services created in the Adapters folder. You can also control security settings for the Web services by using the security features of the Business Services folder.

You are now ready to define new targets to PeopleSoft.

Establishing a Connection (Target) for PeopleSoft

Part of the application definition includes adding a target for OracleAS Adapter for PeopleSoft. Setting up the target in Application Explorer requires information that is specific to the target.

To browse PeopleSoft business objects, you must create a target for the system you intend to use. The target serves as your connection point and is automatically saved after you create it. You must establish a connection to this system every time you start Application Explorer or after you disconnect from the system.

When you launch Application Explorer, the left pane displays (as nodes) the application systems supported by Application Explorer, based on the adapters that are installed.

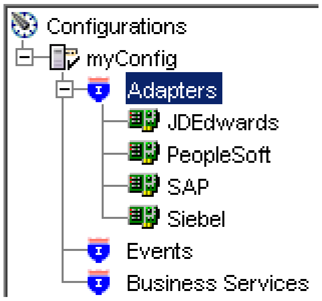
Defining a Target to PeopleSoft

To connect to PeopleSoft for the first time, you must define a new target. OracleAS Adapter for PeopleSoft supports PeopleSoft standard security, in addition to component interface-level security. Once connected to the PeopleSoft application server, application security is managed by user ID, roles and privileges. For more information on PeopleSoft application security, see the appropriate PeopleSoft documentation.

To define a target:

1. In the left pane, expand the **Adapters** node.

   The applications systems supported by Application Explorer appear as nodes based on the adapters that are installed.



2. Right-click the **PeopleSoft** node and select **Add Target**.

   The Add Target dialog box is displayed. Provide the following information:

   a. In the **Name** field, enter a descriptive name for the target, for example, **PSoftTarget**.

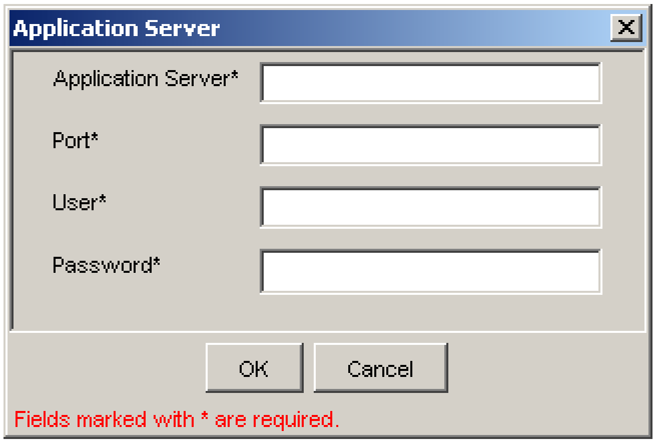
   b. In the **Description** field, enter a description for the target (optional).

   c. From the **Target Type** list, select **Application Server**.

      This is the only possible value for target type.

3. Click **OK**.

   The Application Server dialog box is displayed. You must specify connection information for PeopleSoft and the application server that is hosting PeopleSoft.



Provide the following information:

a. In the **Application Server** field, enter the host name or IP address for the computer that is hosting the PeopleSoft application.

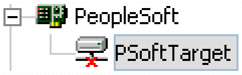
b. In the **Port** field, enter the port number where the PeopleSoft application is listening.

c. In the **User** field, enter a valid user ID for the PeopleSoft application.

d. In the **Password** field, enter a valid password for the PeopleSoft application.

4. Click **OK**.

   In the left pane, the new target (**PSoftTarget**) appears the PeopleSoft node.



You are ready to connect to your PeopleSoft target.

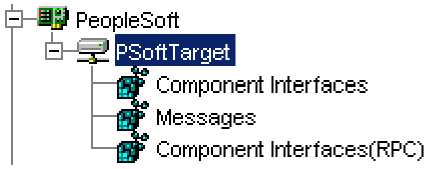
Connecting to a Defined PeopleSoft Target

To connect to an existing target:

1. In the left pane, expand the **Adapters** node.
2. Expand the **PeopleSoft** node.
3. Click the target name (for example, **PSoftTarget**) under the PeopleSoft node.

   The Connection dialog box displays the values you entered for connection parameters.
4. Verify your connection parameters.
5. Provide the correct password.
6. Right-click the target name and select **Connect**.

The x icon disappears, indicating that the node is connected. A list of PeopleSoft business objects is displayed.



Managing a Target

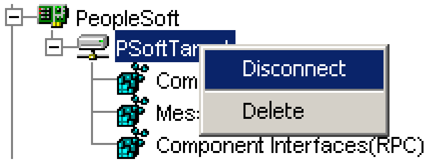
Although you can maintain multiple open connections to different transaction processing systems, it is recommended that you disconnect from connections not in use. After you disconnect, you can modify an existing target.

You can modify the connection parameters when your system properties change. You also can delete a target. The following procedures describe how to disconnect from a target, edit a target, and delete a target.

Disconnecting from a Connection to PeopleSoft

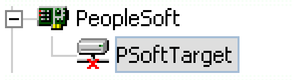
To disconnect from a connection to PeopleSoft:

1. Expand the **Adapters** node.
2. Expand the **PeopleSoft** node.
3. Right-click the target to which you are connected (for example, PSoftTarget), and select **Disconnect**.



Disconnecting from PeopleSoft drops the connection with PeopleSoft, but the node remains.

The x icon appears, indicating that the node is disconnected.

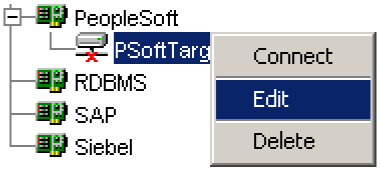


Modifying Connection Parameters

After you create a target for PeopleSoft using Application Explorer, you can edit any of the information that you provided previously.

To edit a target:

1. Verify that the target you want to edit is disconnected.
2. Right-click the target and select **Edit**.



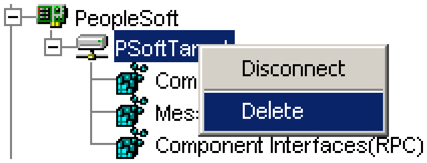
The Application Server dialog box displays the target connection information.

3. Change the properties in the dialog box as required and click **OK**.

Deleting a Connection to PeopleSoft

To delete a connection to PeopleSoft:

1. Locate the target you want to delete.
2. Right-click the target (for example, PsoftTarget), and select **Delete**.



The node disappears from the list of available connections.

Viewing Application System Objects

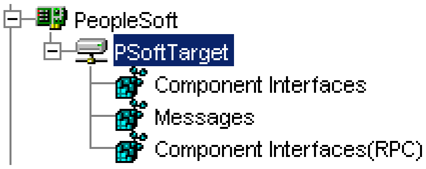
After you are connected to PeopleSoft, Application Explorer enables you to explore and browse business object metadata. For example, Application Explorer enables you to view PeopleSoft Component Interface and Message metadata stored in the PeopleSoft business object repository.

For Component Interfaces(RPC), the adapter enables Delete, Insert, Query, Update, and Find.

To view application system objects:

1. Click the icon to the left of the target name, for example, PSoftTarget.

   The target expands and the available system objects are displayed.



2. To expand the desired PeopleSoft repository node, click the icon to the left of the repository name, for example, Component Interfaces.

   A list of PeopleSoft Component Interfaces appears. You can now generate schemas.

Creating XML Schemas

After you browse the PeopleSoft business object repository, you can generate XML request and response schemas for the object you wish to use with your adapter.

Creating XML Request and Response Schemas Against BSE

To create XML request and response schemas for a PeopleSoft Component Interface against a BSE implementation:

1. Expand the PeopleSoft node and then, expand the **Component Interfaces** node.
2. From the list of **Component Interfaces**, select **LOCATION**.

3. Click the **Request Schema** or **Response Schema** tab to view the request schema information.

   The schema you selected appears.

   After you browse the Component Interfaces and make a selection, the request and response XML schemas are automatically created for that Component Interface and stored in the repository you created.

Creating XML Request and Response Schemas Against the OraclesAS Adapter J2CA

To create XML request and response schemas for a PeopleSoft Component Interface against an OraclesAS Adapter J2CA implementation:

1. Expand the **PeopleSoft** node and then, expand the **Component Interfaces** node.
2. From the list of **Component Interfaces**, select **LOCATION**.
3. Click the **Request Schema** or **Response Schema** tab to view the request schema information.

   The schema you selected appears.

   After you browse the Component Interfaces and make a selection, the request and response XML schemas are automatically created for that Component Interface and stored in the repository you created.

Generating WSDL (J2CA Configurations Only)

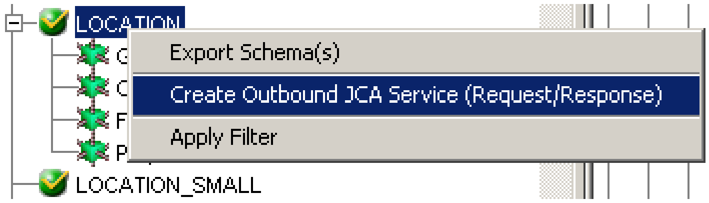
The Web Service Definition Language (WSDL) description of a service enables you to make the service available to other services within a host server. You use Application Explorer to create both request-response (outbound) and event notification (inbound) JCA services of the adapter.

> **Note:** The **Create Inbound JCA Service (Event)** option is only available when the selected node supports events.

To generate a WSDL file for request-response service:

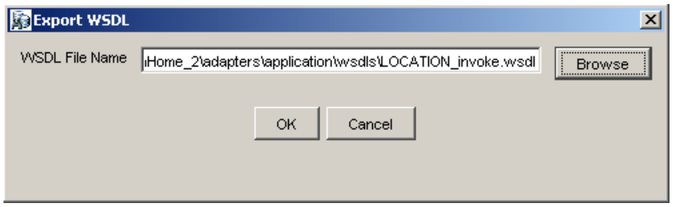
1. After you create a schema, right-click the respective object.

   The following menu is displayed.



2. Select **Create Outbound JCA Service (Request/Response)**.

The Export WSDL dialog box is displayed.



WSDL File Name: \Home_2\adapters\application\wsdls\LOCATION_invoke.wsdl

3. Accept the default name and location for the file.

   The **.wsdl** file extension is added automatically. By default, the names of WSDL files generated for request-response services end with `_invoke`, while those generated for event notification end with `_receive`.

> **Note:** You can organize your WSDL files in subfolders, creating your own WSDL hierarchy structure. Create the folders under `*OracleAS_home*\adapters\application\wsdls\`. The WSIL browser in JDeveloper will display the full tree structure of your WSDL hierarchy.

4. Click **OK**.

   The WSDL file is saved in the specified location.

The procedure for generating WSDL for event notification is similar to request-response. To generate WSDL for event notification, you must first create a channel for every event. See "Generating WSDL for Event Notification" on page 5-13 for a detailed example.

Creating and Testing a Web Service (BSE Configurations Only)

You can create Web services (also known as a **business service**) using Application Explorer. The PeopleSoft Component Interface called LOCATION is used as an example in the following procedure.

> **Note:** In a J2EE Connector Architecture (J2CA) implementation of adapters, Web services are not available. When the adapters are deployed to use OracleAS Adapter for J2CA, the Common Client Interface provides integration services using the adapters.

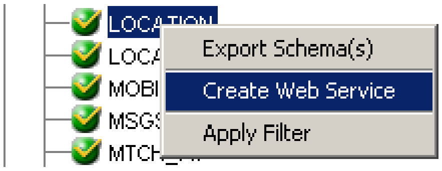
Creating a Web Service

To create a Web service, perform the following steps:

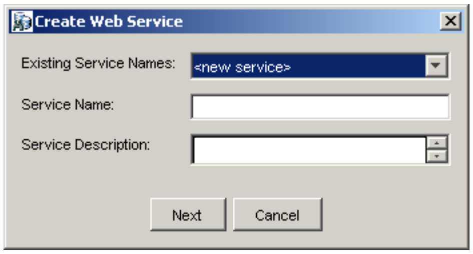
1. Expand the **PeopleSoft** node and then the **Component Interfaces** node.

2. From the list of **Component Interfaces**, select **LOCATION**.

3. Right-click the node from which you want to create a **business service** and select **Create Web Service**.



The Create Web Service dialog box is displayed.



Existing Service Names: <new service>

Service Name:

Service Description:

You can add the business function as a method for a new Web service or as a method for an existing one. Perform the following steps:

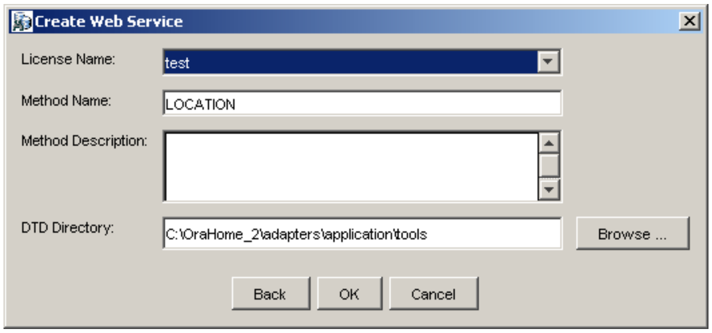
a. From the **Existing Service Names** list, select either **<new service>** or an existing service.

b. If you are creating a new service, specify a service name. This name identifies the Web service in the list of services under the **Business Services** node.

c. Enter a brief description for the service (optional).

4. Click **Next**.

   The license and method dialog box is displayed.



License Name: test

Method Name: LOCATION

Method Description:

DTD Directory: C:\OraHome_2\adapters\application\tools

Perform the following steps:

a. In the **License Name** field, select one or more license codes to assign to the Web service. To select more than one, hold down the **Ctrl** key and click the licenses.

b. In the **Method Name** field, enter a descriptive name for the method.

c. In the **Method Description** field, enter a brief description of the method.

d. The **DTD Directory** field specifies a destination for your Web service. You may click **Browse** to change the default location.

5. Click **OK**.

   Application Explorer switches the view to the **Business Services** node, and the new Web service appears in the left pane.

Testing a Web Service

After a Web service is created, you can test it to ensure that it functions properly. A test tool is provided for testing the Web service.

To test a business service:

1. Click the **Business Services** node to access your Web services.

2. Expand the **Services** node.
3. Select the name of the business service you want to test.

   The business service name appears as a link in the right pane.
4. In the right pane, click the named business services link.

   The test option appears in the right pane. If you are testing a Web service that requires XML input, an input field appears.
5. Enter the appropriate input.
6. Click **Invoke**.

   Application Explorer displays the results.

Identity Propagation

If you test or execute a Web service using a third party XML editor, for example XMLSPY, the Username and Password values that you specify in the SOAP header must be valid and are used to connect to PeopleSoft. The user name and password values that you provided for PeopleSoft during target creation using Application Explorer are overwritten for this Web service request. The following is a sample SOAP header that is included in the WSDL file for a Web service:

```
<SOAP-ENV:Header>
  <m:ibsinfo xmlns:m="urn:schemas-iwaysoftware-com:iwse">
    <m:service>String</m:service>
    <m:method>String</m:method>
    <m:license>String</m:license>
    <m:disposition>String</m:disposition>
    <m:Username>String</m:Username>
    <m:Password>String</m:Password>
    <m:language>String</m:language>
  </m:ibsinfo>
</SOAP-ENV:Header>
```

You can remove the `<m:disposition>` and `<m:language>` tags from the SOAP header, since they are not required.

Configuring an Event Adapter

Events are generated as a result of activity in an application system. You can use events to trigger an action in your application. For example, PeopleSoft may generate an event when customer information is updated. If your application performs an action when this happens, your application is a consumer of this event.

After you create a connection to your application system, you can add events using Application Explorer. To create an event, you must create a port and a channel.

Note: If you are using a J2CA configuration, you must create a new channel for every event and select this channel when you generate WSDL. Additionally, you are not required to create or configure ports for use with BPEL Process Manager. See "J2CA Filtering Port Option for Integration with BPEL Process Manager" on page 2-20 for details on using ports under a J2CA configuration.

- A **port** associates a particular business object exposed by the adapter with a particular disposition. A disposition is a URL that defines the protocol and location of the event data. The port defines the end point of the event consumption. See "Creating and Editing an Event Port" on page 2-18 for more information.

- A **channel** represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by the adapter. See "Creating and Editing a Channel" on page 2-20 for more information.

> **Note:** Oracle Containers for J2EE (OC4J) currently conforms to J2CA 1.0, which does not call for event capabilities. When conforming to J2CA 1.0, only service interactions are supported.

Creating and Editing an Event Port

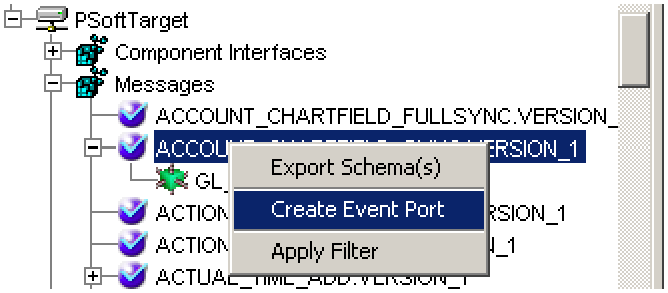
Application Explorer enables you to create event ports from the Adapters node or from the Events node.

> **Note:** You are not required to create event ports for J2CA configurations. You must create event ports for BSE configurations only.

Creating an Event Port from the Adapters Node

You can bypass the Events node and create an event port directly from the Adapters node. Perform the following steps:

1. Right-click any node under **Messages**.
2. Select **Create Event Port**.



The Create Event Port dialog is displayed. Perform the following steps:

a. Enter a name for the event port and provide a brief description.

b. From the **Protocol** drop-down list, select the required disposition, for example, RMI.

c. Enter the disposition URL.

d. Specify the location of your Web service.

3. Click **OK**.

See "Creating an Event Port from the Events Node" on page 2-19 for information on configuring port dispositions.

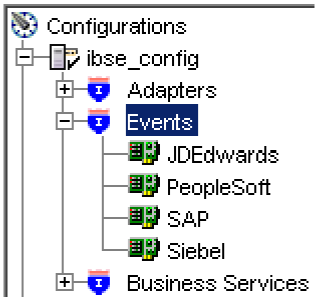
Creating an Event Port from the Events Node

The following procedures describe how to create an event port from the Events node for various dispositions using Application Explorer. You can switch between a BSE and a J2CA deployment by choosing one or the other from the menu in the upper right of Application Explorer.

Creating an Event Port for RMI

To create a specific event port for RMI:

1. Click the **Events** node.



2. Expand the **PeopleSoft** node.
3. Right-click the **Ports** node and select **Add Port**.

   The Add Port dialog box is displayed. Perform the following steps:

   a. Enter a name for the event port and provide a brief description.

   b. From the **Protocol** list, select **RMI**.

   c. In the **URL** field, specify a destination file to which the event data is written using the following format:

   rmi://*host*:*port*;RemoteObject=*APPNAME*;errorTo=*pre-defined port name or another disposition url*

   The following table defines the parameters for the disposition.

Parameter	Description
host	The host name or IP address from which the RMI server accepts RMI requests. If you omit this attribute, the RMI server will accept RMI requests from any host.
port	The port number on which the RMI server listens for RMI requests.
RemoteObject	A home or Enterprise JavaBeans (EJB) object.
errorTo	Predefined port name or another disposition URL to which error logs are sent.

   d. Click **OK**.

The port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created.

You can now associate the event port with a channel. See "Creating and Editing a Channel" on page 2-20 for more information.

Editing an Event Port

To edit an event port:

1. In the left pane, select the event port you want to edit.
2. Right-click the port and select **Edit**.

   The Edit Port pane is displayed.
3. Make the required changes and click **OK**.

Deleting an Event Port

To delete an event port:

1. In the left pane, select the event port you want to delete.
2. Right-click the port and select **Delete**.

   A confirmation dialog box is displayed.
3. To delete the event port you selected, click **OK**.

   The event port disappears from the list in the left pane.

J2CA Filtering Port Option for Integration with BPEL Process Manager

This feature provides the option to use event ports when using OracleAS Adapter for PeopleSoft with BPEL PM under a J2CA configuration. Without this feature, all messages from a particular channel are forwarded to the endpoint and then to the BPEL PM Server. In this case, no schema validation takes place. With the filtering feature, you have the option to associate an event schema to a port that you create and configure in Application Explorer. In this case, at runtime the message is validated against the event schema and only the validated event message is forwarded to the endpoint. If the event message does not match the event schema that is associated to the port, then an error message is written to the log and the event message is ignored.

Creating and Editing a Channel

The following procedures describe how to create a channel for your event. All defined event ports must be associated with a channel.

> **Note:** If using a J2CA configuration, you must create a new channel for every event and select this channel when creating an inbound service. Creating a channel is required for both BSE and J2CA configurations.

Creating a Channel Using Specific Protocols

You can create the following types of channels using Application Explorer:

- HTTP
- TCP
- File

The following procedures explain how to create these channels.

Note: OC4J currently conforms to J2CA 1.0, which does not call for event capabilities. When conforming to J2CA 1.0, only service interactions are supported.

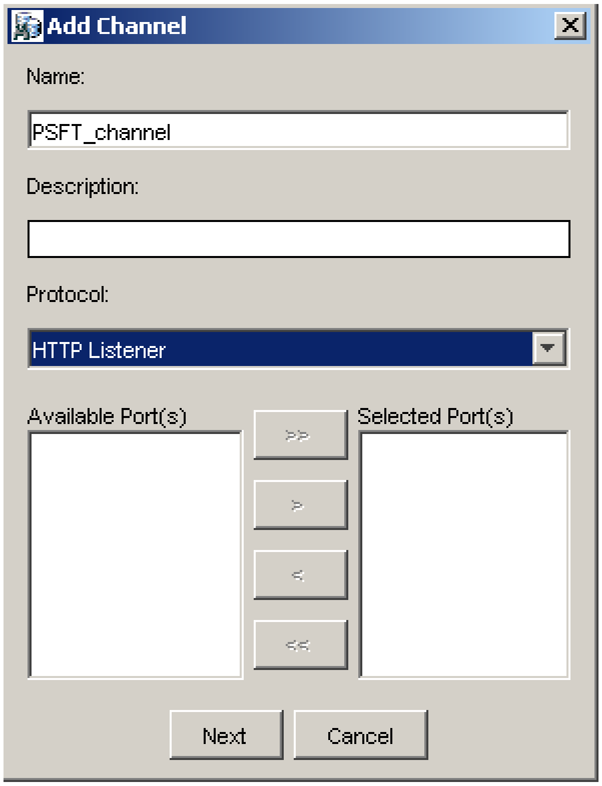
Creating an HTTP Channel

To create an HTTP Channel:

1. Click the **Events** node.
2. Expand the **PeopleSoft** node.

   The ports and channels nodes appear in the left pane.
3. Right-click **Channels** and select **Add Channel**.

   The Add Channel dialog box is displayed.



Add Channel

Name: PSFT_channel

Description:

Protocol: HTTP Listener

   Provide the following information:

   a. Enter a name for the channel, for example, **PSFT_channel**.

   b. Enter a brief description.

   c. From the **Protocol** list, select **HTTP Listener**.

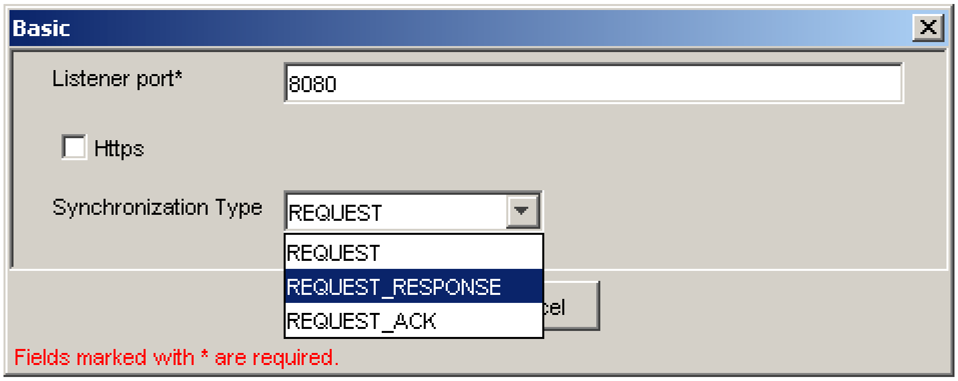
   Important: If you are using BPEL Process Manager with a J2CA configuration without the optional port feature, skip the following step. See "J2CA Filtering Port Option for Integration with BPEL Process Manager" on page 2-20.

   d. Select an event port from the list of available ports.

   To transfer the port to the list of available ports, click the double right **(>>)** arrow. To associate all the event ports, control-click to select each port or click one port and press **Control+A**. Then, click the double right **(>>)** arrow.

4. Click **Next**.

   The Basic dialog box is displayed.



Listener port* 8080

□ Https

Synchronization Type REQUEST

Fields marked with * are required.

5. Enter the system information as specified in the following table:

Parameter	Description
Listener port	Port on which to listen for PeopleSoft event data.
Https	For a secure HTTP connection, select the **Https** check box.
Synchronization Type	Choose from the following synchronization options: ■ REQUEST_RESPONSE ■ REQUEST_ACK **Important:** The PeopleSoft channel will not work if the synchronization type is set to REQUEST.

6. Click **OK**.

   A summary pane is displayed, providing the channel description, channel status, and available ports. All the information is associated with the channel you created.

   The channel appears under the channels node in the left pane.

   An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

   > **Note:** If you are using OracleAS Adapter for PeopleSoft with BPEL Process Manager, do not start the channel, as it is managed by the BPEL PM Server. If you start the channel for testing and debugging purposes, stop it before runtime.

7. Right-click the channel and select **Start**.

   The channel you created becomes active. The X over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select **Stop**.

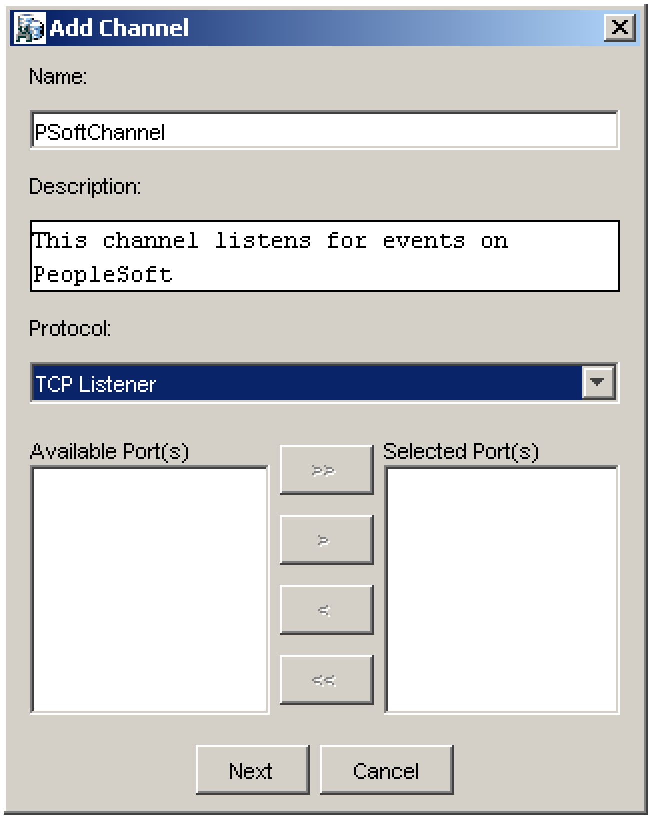
Creating a TCP Channel

1. Click the **Events** node.
2. Expand the **PeopleSoft** node.

The ports and channels nodes appear in the left pane.

3. Right-click **Channels** and select **Add Channel**.

   The Add Channel dialog box is displayed.



Name:

PSoftChannel

Description:

This channel listens for events on PeopleSoft

Protocol:

TCP Listener

   Perform the following steps:

   a. Enter a name for the channel, for example, PSoftChannel.

   b. Enter a brief description.

   c. From the **Protocol** list, select **TCP Listener**.

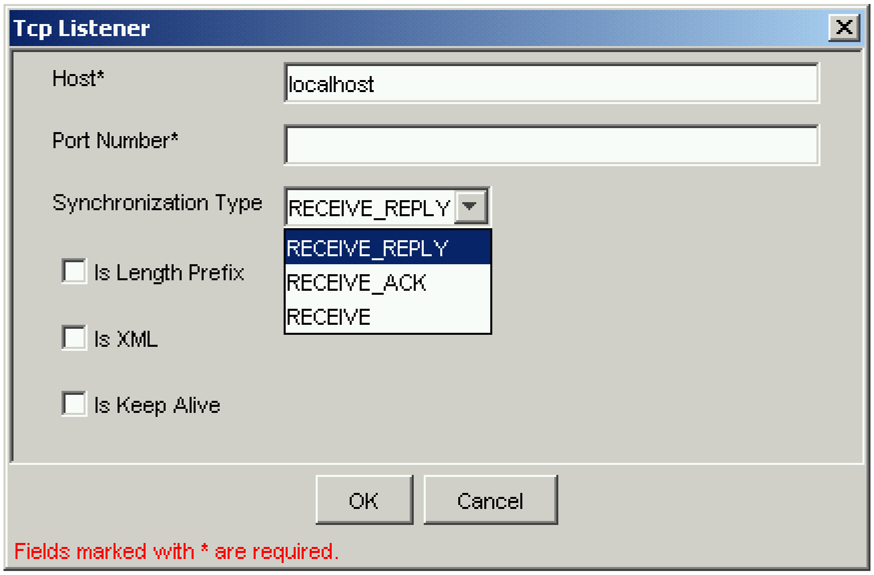
      Important: If you are using BPEL Process Manager with a J2CA configuration without the optional port feature, skip the following two steps. See "J2CA Filtering Port Option for Integration with BPEL Process Manager" on page 2-20.

   d. Select an event port from the list of available ports. To select more than one, hold down the **Ctrl** key and select the ports.

   e. Click >> to transfer the port(s) to the list of selected ports.

4. Click **Next**.

   The Tcp Listener dialog box is displayed.



Tcp Listener

Host* localhost

Port Number*

Synchronization Type RECEIVE_REPLY

□ Is Length Prefix

□ Is XML

□ Is Keep Alive

Fields marked with * are required.

Provide the following information:

a. In the **Host** field, specify the host for your TCP listener (required).

b. In the **Port Number** field, specify the port number for your TCP listener (required).

c. From the **Synchronization Type** list, select from the following synchronization type options:

   RECEIVE_REPLY or

   RECEIVE_ACK

   Important: The PeopleSoft channel will only work with one of these synchronization types.

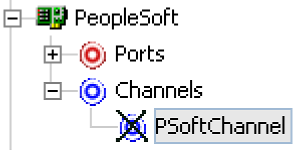
d. **Is Length Prefix** check box: For PeopleSoft events that return data that is not in XML format. The TCP/IP event application must prefix the data with a 4-byte binary length field when writing data to the TCP/IP port.

e. **Is XML** check box: For PeopleSoft events that return data in XML format. No preparser is required.

f. **Is Keep Alive** check box: Maintains continuous communication between the event transaction and the channel.

5. Click **OK**.

   The channel appears under the Channels node in the left pane.



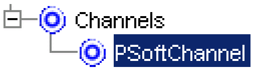
An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

Note: If you are using OracleAS Adapter for PeopleSoft with BPEL Process Manager, do not start the channel, as it is managed by the BPEL PM Server. If you start the channel for testing and debugging purposes, stop it before runtime.

---

6. Right-click the channel node and select **Start**.

   The channel becomes active.



   The X over the icon disappears.

7. To stop the channel, right-click the connected channel node and select **Stop**.

   The channel becomes inactive and an X appears over the icon.

Creating a File Channel

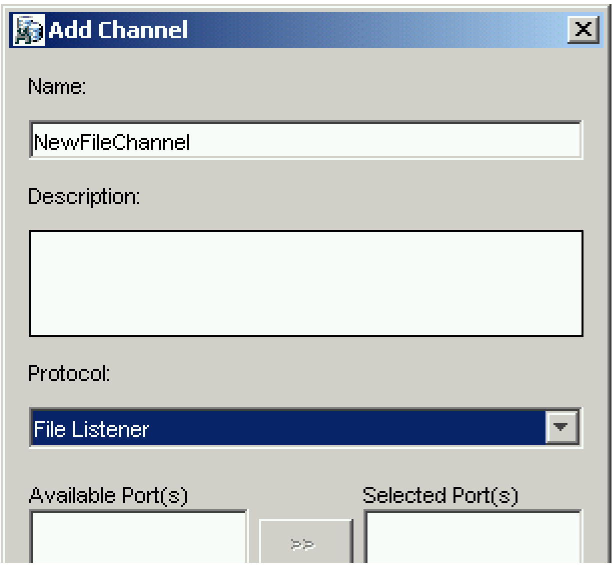
To create a channel for the File listener:

1. Click the **Events** node.
2. In the left pane, expand the **PeopleSoft** node.

   The ports and channels nodes appear.

3. Right-click **Channels** and select **Add Channel**.

   The Add Channel dialog box is displayed.



Name: NewFileChannel

Description:

Protocol: File Listener

   Perform the following steps:

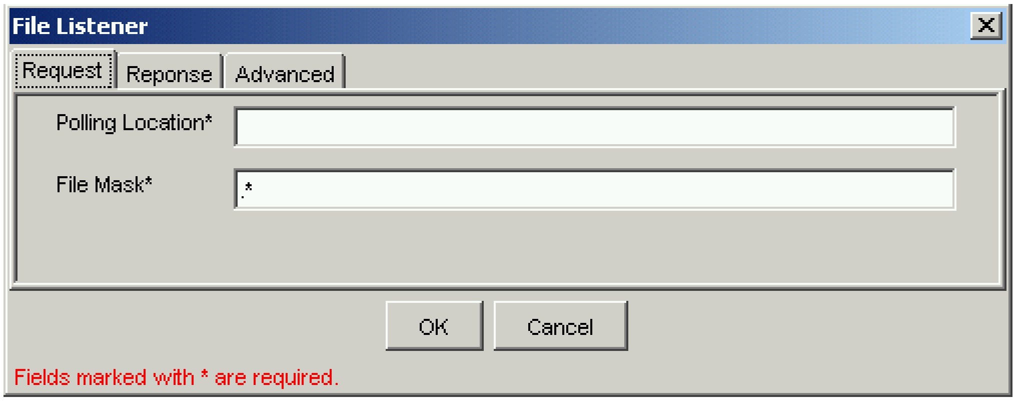
   a. Enter a name for the channel, for example, **NewFileChannel**.
   b. Enter a brief description (optional).
   c. From the **Protocol** list, select **File Listener**.

Important: If you are using BPEL Process Manager with a J2CA configuration without the optional port feature, skip the following step. See "J2CA Filtering Port Option for Integration with BPEL Process Manager" on page 2-20.

d. Select an event port from the list of available ports.

To transfer the port to the list of available ports, click the double right **(>>)** arrow. To associate all the event ports, control-click to select each port or click one port and press **Control+A**. Then, click the double right **(>>)** arrow.

4. Click **Next**.
5. The File Listener dialog box is displayed.



Enter the system information as follows:

a. In the **Request** tab, enter values for the following parameters:

Parameter	Description
Polling Location	Target file system location for the PeopleSoft XML file.
File Mask	File name to be used for the output file generated as a result of the operation.

b. In the **Response** tab, enter values for the following parameters:

Parameter	Description
Synchronization Type	Choose from the following synchronization options: ■ REQUEST_RESPONSE ■ REQUEST_ACK **Important:** The PeopleSoft channel will not work if the synchronization type is set to REQUEST.
Response/Ack Directory	Target file system location for the PeopleSoft XML file.

c. In the **Advanced** tab, enter values for the following parameters:

Parameter	Description
Error directory	Directory to which documents with errors are written.
Poll interval (msec)	Interval (in milliseconds) when to check for new input. The default is three seconds. Optional.

Parameter	Description
Processing Mode	**Sequential** indicates single processing of requests. **Threaded** indicates processing of multiple requests simultaneously.
Thread limit	If you selected threaded processing, indicate the maximum number of requests that can be processed simultaneously.

6. Click **OK**.

   A summary pane is displayed, providing the channel description, channel status, and available ports. All the information is associated with the channel you created.

   The channel appears under the channels node in the left pane. An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

   > **Note:** If you are using OracleAS Adapter for PeopleSoft with BPEL Process Manager, do not start the channel, as it is managed by the BPEL PM Server. If you start the channel for testing and debugging purposes, stop it before runtime.

7. Right-click the channel and select **Start**.

   The channel you created becomes active. The X over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select **Stop**.

Editing a Channel

You must stop the channel before you can edit it. To edit a channel:

1. In the left pane, locate the channel you want to edit.
2. Right-click the channel and select **Edit**.

   The Edit channels pane is displayed.

3. Make the required changes to the channel configuration and click **Finish**.

Deleting a Channel

You must stop the channel before you can delete it. To delete a channel:

1. In the left pane, locate the channel you want to delete.
2. Right-click the channel and select **Delete**.

   A confirmation dialog box is displayed.

3. To delete the channel you selected, click **OK**.

   The channel disappears from the list in the left pane.

3

OC4J Deployment and Integration

This chapter describes Oracle Containers for J2EE (OC4J) deployment and integration with OracleAS Adapter for PeopleSoft.

This chapter discusses the following topics:

- Adapter Integration with OC4J
- Deployment of Adapter
- Updating Adapter Configuration
- How to Write a Java Application Client Using the CCI API

See Also:

- *Oracle Application Server Adapter Concepts*

Adapter Integration with OC4J

OracleAS Adapter for PeopleSoft is deployed within an OC4J container during installation. All client applications run within the OC4J environment. In J2CA deployment, the Common Client Interface (CCI) integrates an OC4J client application with a resource adapter.

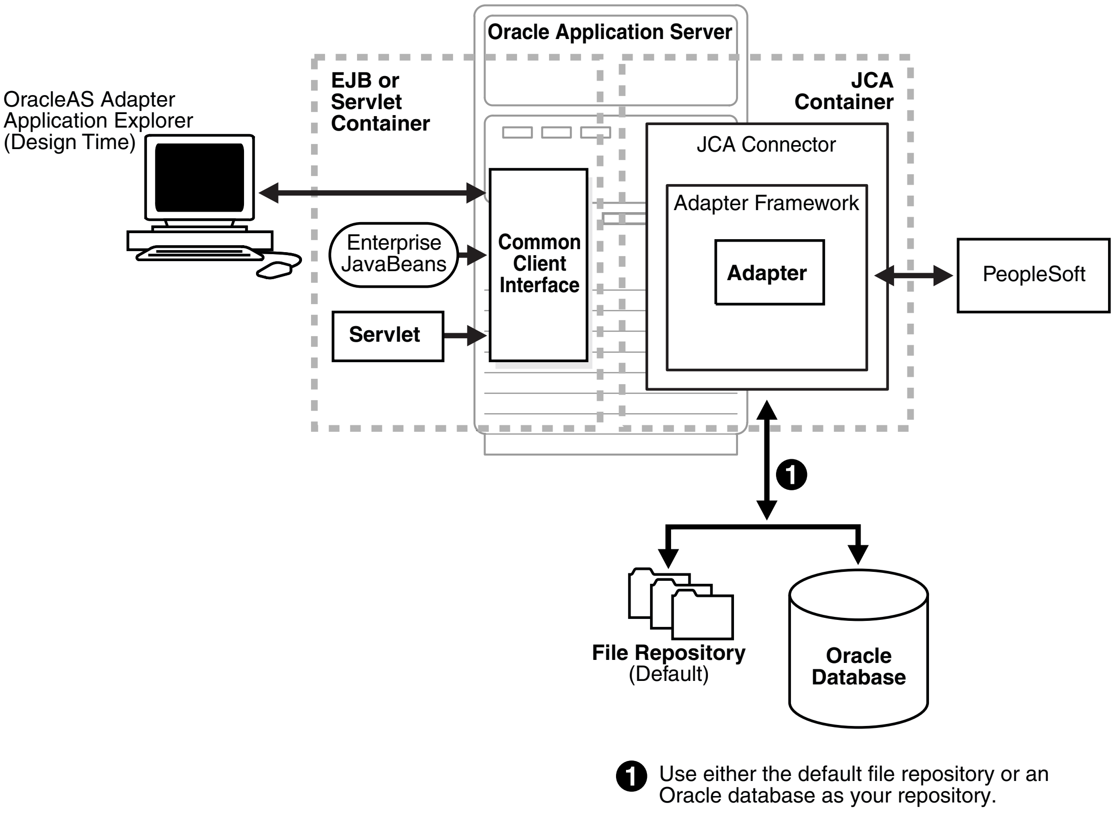
See Also:

- Oracle Application Server Adapters Integration with OC4J" in *Oracle Application Server Adapter Concepts*

Deployment of Adapter

Figure 3–1 shows deployment of the Connector to the Oracle Application Server. In a runtime service scenario, an EJB, servlet, or Java program client makes CCI calls to J2CA resource adapters. The adapters process the calls as requests and send them to the EIS. The EIS response is then sent back to the client.

Figure 3–1 Oracle Application Server Adapter J2CA Architecture


Oracle Application Server
EJB or Servlet Container
JCA Container
OracleAS Adapter Application Explorer (Design Time)
JCA Connector
Adapter Framework
Enterprise JavaBeans
Common Client Interface
Adapter
PeopleSoft
Servlet
1
File Repository (Default)
Oracle Database
1 Use either the default file repository or an Oracle database as your repository.

See Also: *Oracle Application Server Adapter Concepts*

Updating Adapter Configuration

During the J2CA deployment of OracleAS Adapter for PeopleSoft, OC4J generates a deployment descriptor called `oc4j-ra.xml`, located in *`OC4J_home`*`\integration\orabpel\system\appserver\oc4j\j2ee\home\application-deployments\default\iwafjca`.

Note: Your installation contains more than one file named `oc4j-ra.xml`. The OC4J deployment descriptor described in this section is located in the directory specified earlier.

Creating a Managed Connector Factory Object

The `oc4j-ra.xml` descriptor provides OC4J-specific deployment information for resource adapters. For example, the default `jca_sample` configuration in Application Explorer is represented in the `oc4j-ra.xml` file as follows:

```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD
Oracle Connector 9.04//EN"
"http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

The parameters defined in the oc4j-ra.xml file are described in the following table:

Parameter Name	Description
IWayHome	The base installation directory for the OracleAS packaged application adapter.
IWayConfig	The adapter configuration name as defined in Application Explorer. For example, OracleAS Adapter for PeopleSoft has a preconfigured jca_sample configuration in Application Explorer.
IWayRepoURL	The URL to use when opening a connection to the database. This is necessary only when using an Oracle database as the BSE repository. See "Configuring BSE System Settings" on page 2-3 for more information.
IWayRepoUser	User name to use when connecting to the database. This is necessary only when using an Oracle database as the BSE repository. See "Configuring BSE System Settings" on page 2-3 for more information.
IWayRepoPassword	Password. If provided, it overwrites configuration. This is necessary only when using an Oracle database as the BSE repository. See "Configuring BSE System Settings" on page 2-3 for more information.
loglevel	It overwrites the level set by the ManagedConnectionFactory property. .

Creating Multiple Managed Connector Factory Objects

To establish multiple managed connector factory objects, you must edit the oc4j-ra.xml file and add more <connector-factory> nodes. For example, the default jca_sample configuration in Application Explorer is represented in the oc4j-ra.xml file as follows:

```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD
Oracle Connector 9.04//EN"
"http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

To create multiple managed connector factory objects, you must add new <connector-factory> nodes in the file. For example:

```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD
Oracle Connector 9.04//EN"
"http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection1"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
<connector-factory location="eis/OracleJCAAdapter/DefaultConnection2"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample2"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

How to Write a Java Application Client Using the CCI API

The following example shows the code structure for using CCI with packaged application adapters. The code sample is shown in four steps.

Step 1. Obtain the Connection Factory

The connection factory is obtained by JNDI lookup.

```
InitialContext context = new InitialContext();
ConnectionFactory cf = (ConnectionFactory)context.lookup(iwayJndi)
```

Step 2. Obtaining a Connection for the Adapter

IWAFConnectionSpec is an implementation of ConnectionSpec used for creating a design time or runtime service adapter connection. The ConnectionSpec has seven parameters. Connection Pooling is fully supported and established based on these parameters, except log level.

Parameter Name	Description
adapterName	Name of the packaged application adapter.
config -	Adapter configuration name. NOT REQUIRED FOR IWAEAdapter.
language	Default is en.
country	Default is us.
userName	User name. If provided, it overwrites configuration.

Parameter Name	Description
password	Password. If provided, it overwrites configuration.
logLevel	It overwrites the level set by the ManagedConnectionFactory property.

A connection pool is a set of client connections to a specific destination. The pool may automatically create new connections to the specified remote system or return an already existing connection. It also provides methods to return a connection back to the pool when it is no longer required.

A connection pool can check which connections are no longer in use and can be closed to save system resources. The time period after which the pool checks the connections and the time after which a connection will time out can be configured by the calling application.

A pool is always bound to one user ID and password, meaning that all connections taken from this pool will also use these credentials. A PeopleSoft connection is always bound to a PeopleSoft user ID and a PeopleSoft Client number.

If you log on with a pool size that is set to one, no connection pool is created (one user ID and one process thread). If you log on with a pool size that is greater than one, a pool is created with a size of n, where n is the number you specified.

Note: Currently the OracleAS Adapter J2CA supports only basic security mapping. The DEBUG log level provides detailed information on the mapping behavior. It functions as follows:

- If the user name and password are not set and no security is provided by the application server, the OracleAS Adapter J2CA will still let it pass and rely on the adapter configuration security information.
- If the user name and password are set, these values will overwrite the adapter configuration. The OracleAS Adapter J2CA compares this information with the security information provided by the application server and log in case the values do not match. However, it still allows the information through.

The iWAFConnectionSpec can be set to initiate an interaction with PeopleSoft by specifying the adapter name and configuration parameters in the ConnectionSpec. For example,

```
iWAFConnectionSpec cs = new IWAFConnectionSpec();
  cs.setAdapterName(ADAPTER);
  cs.setConfig(TARGET);
  cs.setLogLevel(LOG_LEVEL);  // Adapter layer log level
  Connection c = cf.getConnection(cs);// where cf is the connection factory
```

In this snippet, `ADAPTER` and `TARGET` refer to the adapter being deployed, in this case PeopleSoft, and the name of a target defined in Application Explorer. For more information, see "Complete Code Sample" on page 3-6.

Step 3. Create Interaction with InteractionSpec for runtime

```
Interaction i = c.createInteraction();
  IWAFInteractionSpec is = new IWAFInteractionSpec();
  is.setFunctionName(IWAFInteractionSpec.PROCESS);
```

Two functions can be set: PROCESS and IWAE. PROCESS is used at runtime. IWAE is used when you are using the IAEAdapter at design time.

Step 4. Create Input Record and Execute Interaction

In this case, to complete the EIS invocation, a PeopleSoft message is referenced. The schema is provided by Application Explorer.

A standard J2CA indexed record is used in this example:

```
// Use JCA IndexRecord, named "input" for runtime processing.
IndexedRecord rIn = cf.getRecordFactory().createIndexedRecord("input");
rIn.add(msg_run);
  IndexedRecord rOut = (IndexedRecord)i.execute(is, rIn);
System.out.println((String)rOut.get(0));
```

A special record is supported in this example:

```
//IWAFRecord rIn = new IWAFRecord("input");
//rIn.setRootXML(msg_run);
//IWAFRecord response = executeRunInteraction(c, rIn);
  //IWAFRecord rOut = (IWAFRecord)i.execute(is, rIn);
//System.out.println(rOut.getRootXML());
```

Where `msg_run` is an instance XML document generated from the schema created by Application Explorer. For example, the following is a sample PeopleSoft request XML document.

```
<?xml version="1.0" encoding="UTF-8" ?>
<PS8>
  <component perform="browse">LOCATION</component>
  <key name="Setid">SHARE</key>
  <key name="Location">ALBERTA</key>
</PS8>
```

Complete Code Sample

The following is a sample of the complete code:

```
import javax.resource.cci.*;
import com.ibi.afjca.cci.*;
import com.ibi.afjca.spi.*;

/**
 * The purpose of this sample is to illustrate how to use the IWAF Universal
 * JCA connector.
 */
public class IWAFJCASimple {

private static String  HOME    = "c:/iway/xfoc/components/iwafcont/dist";
private static String  CONFIG    = "base";
private static String  LOG_LEVEL = "FATAL";

private static String ADAPTER = "PeopleSoft";
private static String TARGET  = "PeopleSoft_connection";

// Input Message
private static String msg_run = "<PeopleSoft/>";

  public static void main(String[] args) throws Exception {
```

```
// 1. Getting the Connection factory through JNDI lookup
// ---------------------------------------------------------
  InitialContext context = new InitialContext();
  ConnectionFactory cf = (ConnectionFactory)context.lookup(iwayJndi)
  // 2. Getting a connection for a particular adapter target, in this case
PeopleSoft
// ---------------------------------------------------------
  IWAFConnectionSpec cs = new IWAFConnectionSpec();
  cs.setAdapterName(ADAPTER);
  cs.setConfig(TARGET);
  cs.setLogLevel(LOG_LEVEL);  // Adapter layer log level
  Connection c = cf.getConnection(cs);// where cf is the connection factory

  // 3. Create interaction with interactionSpec for RUNTIME
// ---------------------------------------------------------
  Interaction i = c.createInteraction();
  IWAFInteractionSpec is = new IWAFInteractionSpec();
  is.setFunctionName("PROCESS");

  // 4. Create input Record and execute interaction
// ---------------------------------------------------------

  // 4.1 Using JCA standard Indexed Record
// Use JCA IndexRecord, named "input" for runtime processing.
IndexedRecord rIn = cf.getRecordFactory().createIndexedRecord("input");
rIn.add(msg_run);
  IndexedRecord rOut = (IndexedRecord)i.execute(is, rIn);
System.out.println((String)rOut.get(0));

  // 4.2 Our own Record is supported here
//IWAFRecord rIn = new IWAFRecord("input");
//rIn.setRootXML(msg_run);
//IWAFRecord response = executeRunInteraction(c, rIn);
  //IWAFRecord rOut = (IWAFRecord)i.execute(is, rIn);
//System.out.println(rOut.getRootXML());

  } // main()

}
```

4 Integration with BPEL Process Manager

OracleAS Adapter for PeopleSoft integrates seamlessly with Business Process Execution Language (BPEL) Process Manager to facilitate Web service integration. BPEL Process Manager is based on the Service-Oriented Architecture (SOA). It consumes adapter services exposed as Web Service Definition Language (WSDL) documents.

This chapter includes the following topics:

Overview of Adapter Integration with BPEL Process Manager

To integrate with BPEL Process Manager, OracleAS Adapter for PeopleSoft must be deployed in the same OC4J container as BPEL Process Manager. The underlying adapter services must be exposed as WSDL files, which are generated during design time in Oracle Application Server Adapter Application Explorer (Application Explorer) for both request-response (outbound) and event notification (inbound) services of the adapter. See "Generating WSDL (J2CA Configurations Only)" on page 2-14 for more information.

The generated WSDL files are used to design the appropriate BPEL processes for inbound or outbound adapter services. A completed BPEL process must be successfully compiled in a BPEL designer and deployed to a BPEL server. Upon deployment to the BPEL server, every newly built process is automatically deployed to the Oracle BPEL Console, where you run, monitor, and administer BPEL processes, as well as listen to adapter events.

When using the adapter with BPEL Process Manager installed on OracleAS Middle Tier, your middle-tier BPEL PM home directory is OC4J_BPEL, located as follows:

```
OracleAS_home\j2ee\OC4J_BPEL
```

See Also:

- *Oracle Application Server Adapter Concepts*
- *Oracle BPEL Process Manager Developer's Guide*

Deployment of Adapter

During installation, OracleAS Adapter for PeopleSoft is deployed as a J2CA 1.0 resource adapter within the OC4J J2CA container. The adapter must be deployed in the same OC4J container as BPEL Process Manager.

See Also: *Oracle Application Server Adapter Concepts*

Design Time

The following tools are required to complete your adapter design-time configuration:

- OracleAS Adapter Application Explorer (Application Explorer)
- Oracle JDeveloper BPEL Designer (JDeveloper) or Eclipse

Note: The examples in this chapter demonstrate the use of JDeveloper.

Before you design a BPEL process, you must create a schema and generate the respective WSDL file using Application Explorer. See "Generating WSDL (J2CA Configurations Only)" on page 2-14 for more information.

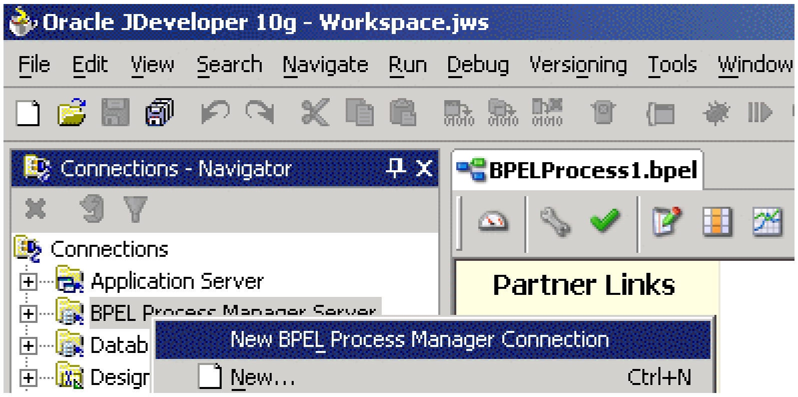
Design a BPEL Process for Request-Response Service (Outbound)

An outbound BPEL process consists of PartnerLink, Invoke, and Assign process activities. You must first create a new BPEL Process Manager connection and a synchronous BPEL process template.

Create a New Connection to BPEL PM Server

To create a new BPEL Process Manager connection:

1. Display the connections by clicking the **Connections** tab at the bottom of the upper left pane in JDeveloper.

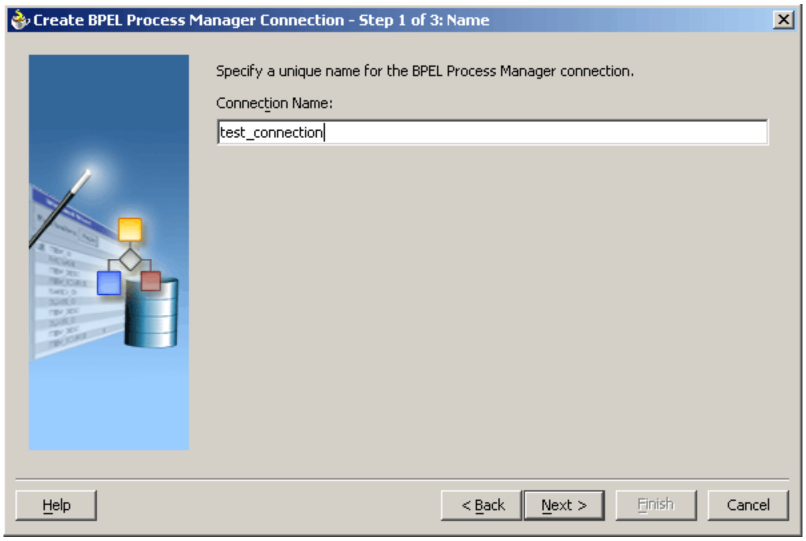


2. Right-click **BPEL Process Manager Server** and select **New BPEL Process Manager Connection**.

   The Create BPEL Process Manager Connection - Welcome dialog box is displayed.

3. Click **Next**.

   The Create BPEL Process Manager Connection dialog box is displayed.



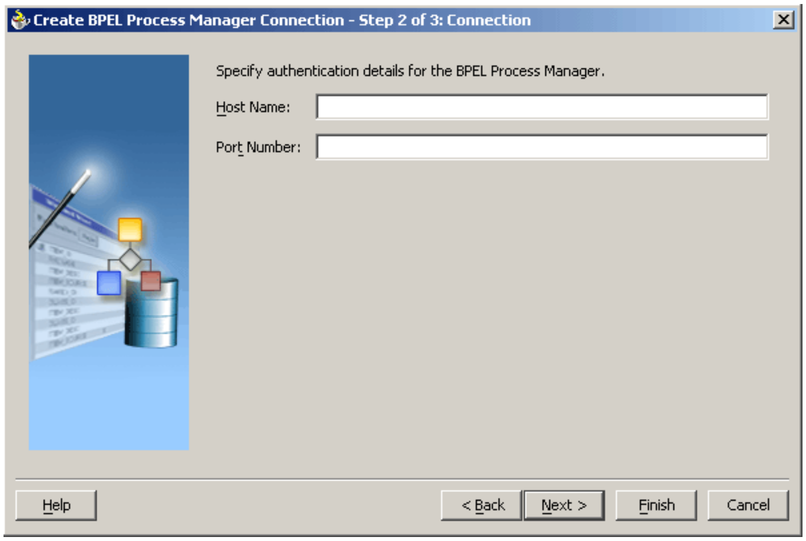
Create BPEL Process Manager Connection - Step 1 of 3: Name

Specify a unique name for the BPEL Process Manager connection.

Connection Name: test_connection

4. Specify a unique name for your BPEL Server connection and click **Next**.

   The Create BPEL Process Manager Connection dialog box is displayed.



Create BPEL Process Manager Connection - Step 2 of 3: Connection

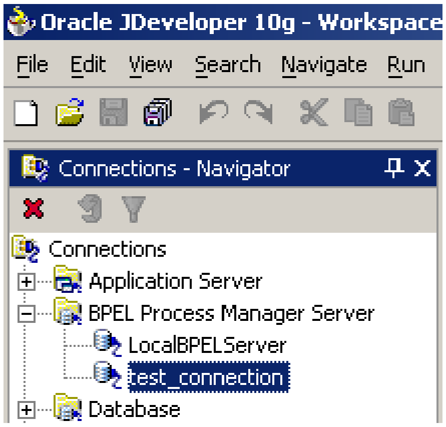
Specify authentication details for the BPEL Process Manager.

Host Name:

Port Number:

5. Specify a valid host name and port number for the BPEL PM Server you wish to connect to.
6. Click **Finish**.

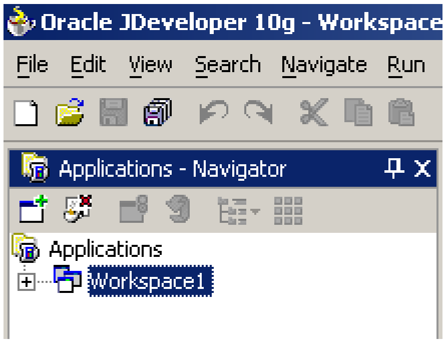
   Your newly created server connection is displayed in the Connections tab under the BPEL Process Manager Server node.



Create a New BPEL Project for Outbound Interaction (Synchronous Process)

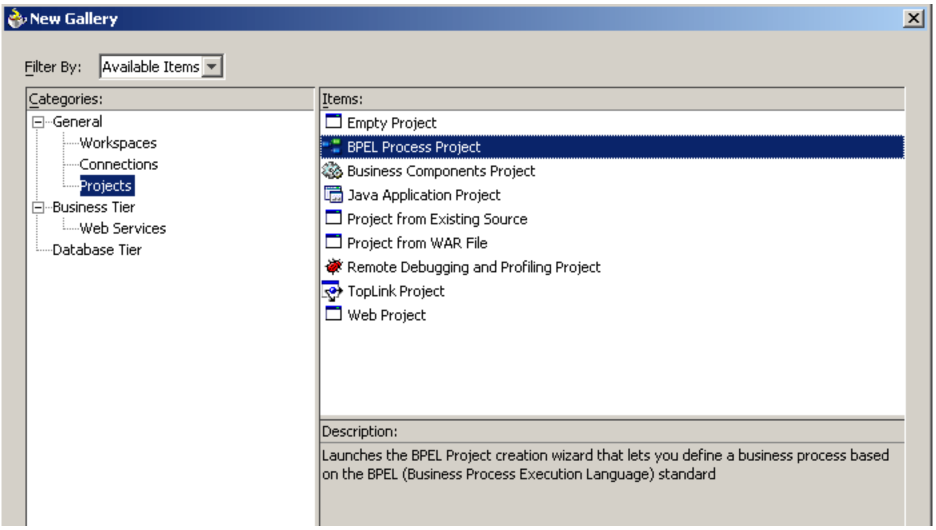
To create a new BPEL project for a synchronous process:

1. At the bottom of the upper left pane, click the **Applications** tab and select a workspace for your project.



2. Right-click the workspace and select **New Project**.

   The New Gallery window is displayed.

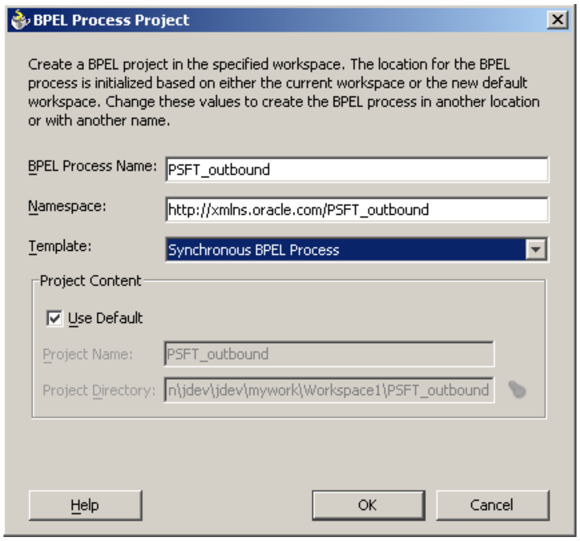


Filter By: Available Items

Description:
Launches the BPEL Project creation wizard that lets you define a business process based on the BPEL (Business Process Execution Language) standard

3. From the Items list, select **BPEL Process Project** and click **OK**.

   The BPEL Process Project dialog box is displayed.



Create a BPEL project in the specified workspace. The location for the BPEL process is initialized based on either the current workspace or the new default workspace. Change these values to create the BPEL process in another location or with another name.

BPEL Process Name: PSFT_outbound

Namespace: http://xmlns.oracle.com/PSFT_outbound

Template: Synchronous BPEL Process

Project Content

☑ Use Default

Project Name: PSFT_outbound

Project Directory: n\jdev\jdev\mywork\Workspace1\PSFT_outbound

4. Perform the following steps:
   - **a.** Specify a name for the BPEL process.

     The Namespace field is updated automatically.
   - **b.** From the Template list, select **Synchronous BPEL Process**.
5. Click **OK**.

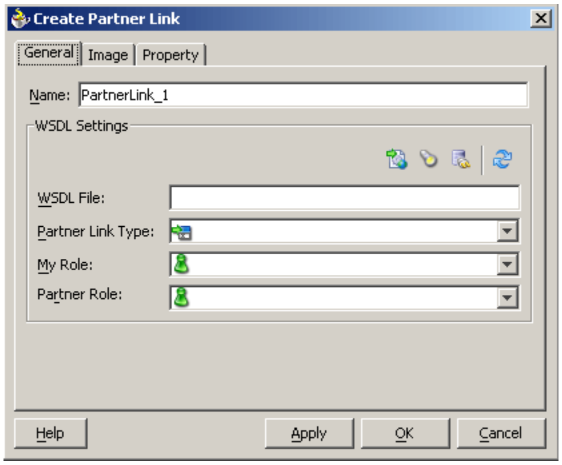
Create an Outbound PartnerLink Activity

When designing a BPEL process, a PartnerLink activity must be created to invoke the PeopleSoft service. A PartnerLink describes a set of operations within a Web service. The WSDL document is the external contract to which the Web service conforms. Given a WSDL, any BPEL process can initiate a Web service through a PartnerLink.

To create an outbound PartnerLink using the WSDL file you generated in Application Explorer:

1. From the Process Activities pane on the right, drag and drop a PartnerLink to the visual editor.

   The Create Partner Link dialog box is displayed.



Name: PartnerLink_1

WSDL Settings

WSDL File:

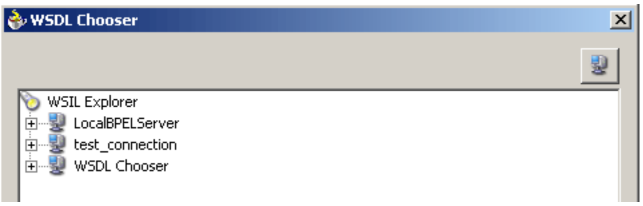
Partner Link Type:

My Role:

Partner Role:

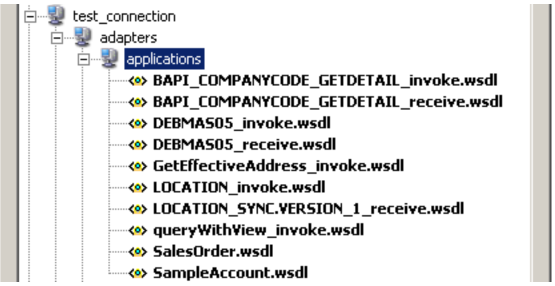
2. Click the **WSIL browser** icon (second icon from the left preceding the **WSDL File** field).

   The WSDL Chooser dialog box is displayed.



3. Expand your new connection, then expand **adapters**, and then **applications**.

   The WSDL tree displayed in the WSDL Chooser dialog box lists any WSDL files you have created using Application Explorer. The WSDL tree is generated by a WSDL servlet, which is automatically deployed as part of the BPEL Server installation.



test_connection
adapters
applications
BAPI_COMPANYCODE_GETDETAIL_invoke.wsdl
BAPI_COMPANYCODE_GETDETAIL_receive.wsdl
DEBMAS05_invoke.wsdl
DEBMAS05_receive.wsdl
GetEffectiveAddress_invoke.wsdl
LOCATION_invoke.wsdl
LOCATION_SYNC.VERSION_1_receive.wsdl
queryWithView_invoke.wsdl
SalesOrder.wsdl
SampleAccount.wsdl

> **Note:** If you have organized your WSDL files in subfolders, the WSIL browser will display the full tree structure of your WSDL hierarchy. By default, the names of all WSDL files generated for outbound adapter services end with `_invoke`.

4. Select **LOCATION_invoke.wsdl** and click **OK**.

   The **WSDL File** field in the Create Partner Link dialog box displays the name and location of the selected WSDL file. The **Partner Link Type** field specifies the PartnerLink defined in the WSDL file.

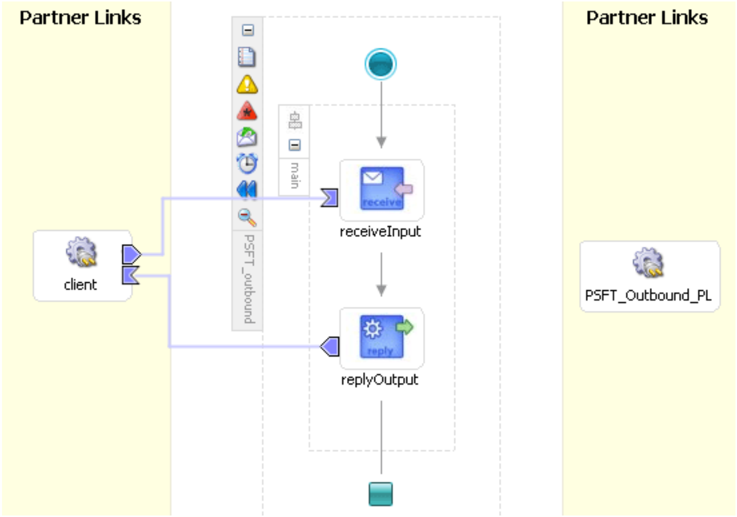
   Perform the following steps:

   a. Leave the **My Role** field unspecified. The role of the PartnerLink is null, as it will be synchronously invoked from the BPEL process.

   b. From the **Partner Role** list, select the default value **LOCATIONRole**. This is the role of the BPEL process.

5. Click **OK**.

   The new PartnerLink appears in the visual editor.



6. Select **Save** from the **File** menu.

Create an Outbound Invoke Activity

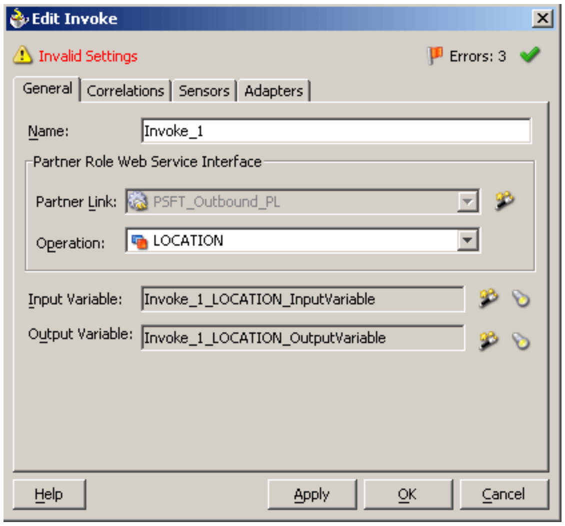
This activity enables you to specify an operation you want to invoke for the service identified by its PartnerLink. The Invoke activity opens a port in the process that is used to send and receive data. It uses this port to submit required data and receive a response. For synchronous callbacks, only one port is needed for both the send and the receive functions.

To create an outbound Invoke activity:

1. From the **Process Activities** pane on the right, drag an **Invoke** activity to the visual editor and place it between the Receive activity (`receiveInput`) and the Reply activity (`replyOutput`).

2. Extend a connection between the Invoke activity and your newly-created PartnerLink.

   The Edit Invoke dialog box is displayed.


Edit Invoke

Invalid Settings

Errors: 3

Name: Invoke_1

Partner Role Web Service Interface

Partner Link: PSFT_Outbound_PL

Operation: LOCATION

Input Variable: Invoke_1_LOCATION_InputVariable

Output Variable: Invoke_1_LOCATION_OutputVariable

Note: Ignore any invalid settings and error warnings.

Perform the following steps:

a. Click the first icon to the right of the **Input Variable** field, then click **OK** in the Create Variable window that is displayed.

   A global variable is automatically created in the Input Variable field.

b. Click the first icon to the right of the **Output Variable** field, then click **OK** in the Create Variable window that is displayed.

   A global variable is automatically created in the Output Variable field.

c. Click **Apply**.

   The Edit Invoke window should no longer display any warnings or errors.

3. Click **OK**.
4. Select **Save** from the **File** menu.

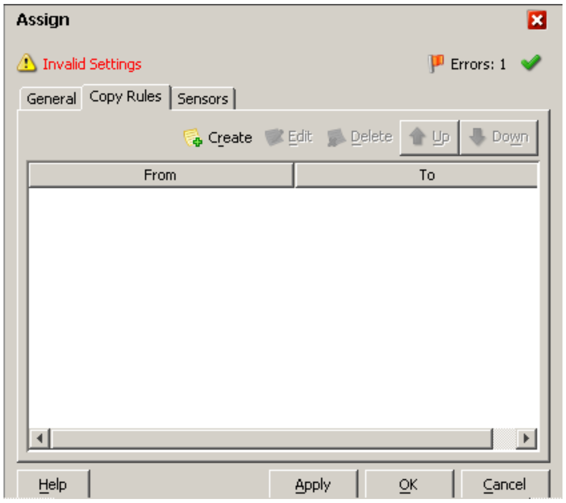
Create an Assign Activity

An Assign activity provides a method for simple data manipulation, such as copying the contents of one variable to another. This Assign activity maps the input variable of the PeopleSoft process to the PeopleSoft PartnerLink input.

To create an Assign activity:

1. From the **Process Activities** pane on the right, drag an **Assign** activity to the visual editor and place it between the Receive activity (`receiveInput`) and the new Invoke activity (`Invoke_1`).
2. Double-click the **Assign** activity icon.

   The Assign dialog box is displayed.



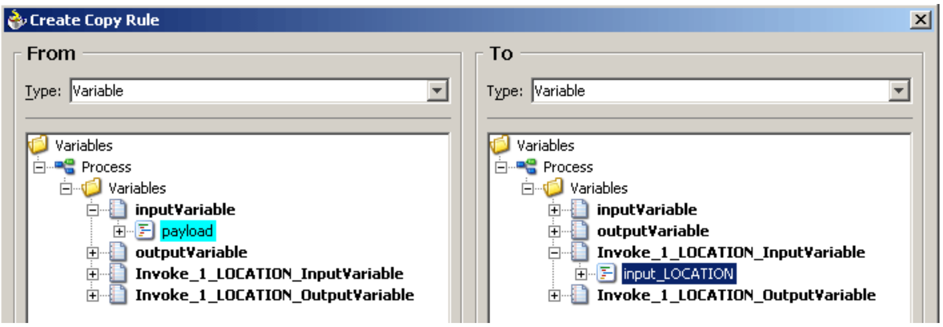
3. In the Copy Rules tab, click **Create**.

   The Create Copy Rule dialog box is displayed.Perform the following steps:

a. In the **From** pane, expand **Variables**, then **inputVariable**, and then highlight **payload**.

b. In the **To** pane, expand **Variables**, then **Invoke_1_LOCATION_InputVariable**, and then highlight **input_LOCATION**.

Your Create Copy Rule dialog box should look as follows:



4. To close the Create Copy Rule dialog box and the Assign dialog box, click **OK**.

Create a Second Assign Activity

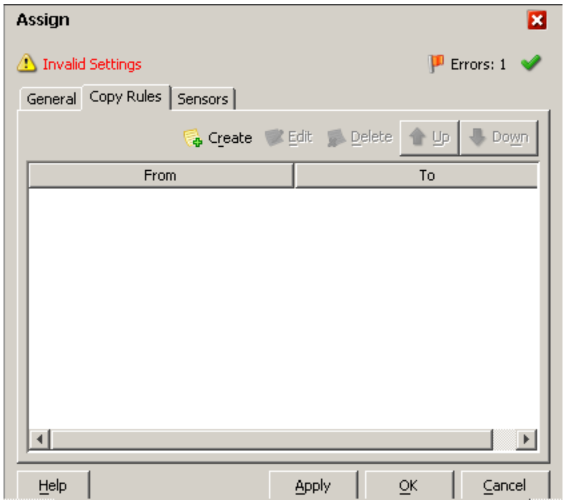
This Assign activity maps the output variable of the PeopleSoft process to the PeopleSoft PartnerLink output.

To create a second Assign activity:

1. From the **Process Activities** pane on the right, drag another **Assign** activity to the visual editor and place it between the Invoke activity (`Invoke_1`) and the Reply activity (`replyOutput`).

2. Double-click the **Assign** activity icon.

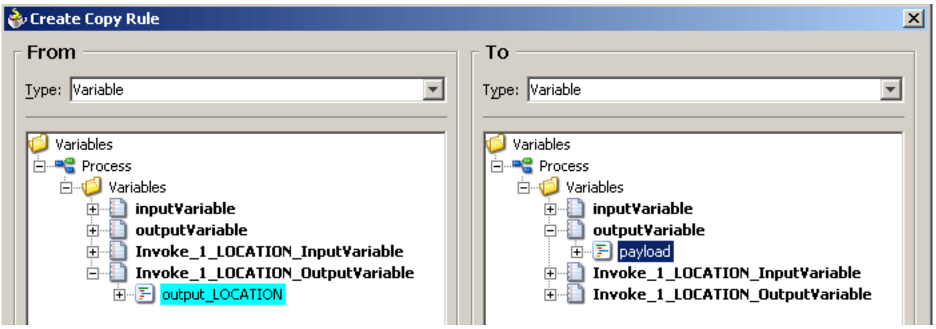
   The Assign settings dialog box is displayed.



Note: Ignore any invalid settings and error warnings.

3. In the Create Copy Rule dialog box, map **Invoke_1_LOCATION_OutputVariable**, **output_LOCATION** to **outputVariable**, **payload**.

   Verify that you have mapped all variables as follows:



4. To close the Create Copy Rule dialog box and the Assign dialog box, click **OK**.
5. Select **Save** from the **File** menu.

You have completed the design of your BPEL process.

See "Invoking Adapter Request-Response Service from BPEL Process Manager" on page 4-15 for information on how to deploy and manage your outbound process.

See Also:

- *Oracle BPEL Process Manager Developer's Guide*
- *Oracle Application Server Adapter Concepts*

Design a BPEL Process for Event Handling (Inbound)

An inbound BPEL process consists of a PartnerLink and a Receive process activity. You must first create a channel and a new BPEL Process Manager Server connection. See Chapter 5, "BPEL Process Manager Integration Examples" for instructions on how to perform these procedures.

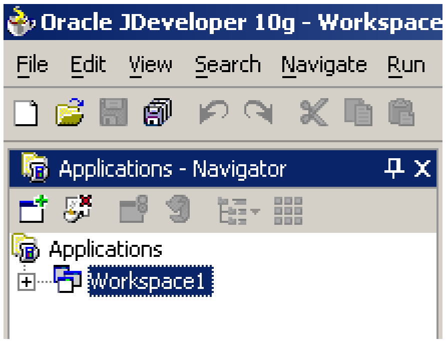
Note: You must create a separate channel for every event and select that channel when you generate WSDL for inbound interaction using Application Explorer. Do not start the channel in Application Explorer, as BPEL Process Manager manages endpoint activation independently. See "PeopleSoft Event Integration" on page 5-11 for more information.

Create a New BPEL Project for Inbound Interaction (Empty Process)

Before you create a BPEL project, verify that your BPEL Server is running. After you have created a new server connection, you are ready to design an empty process template for your BPEL project.

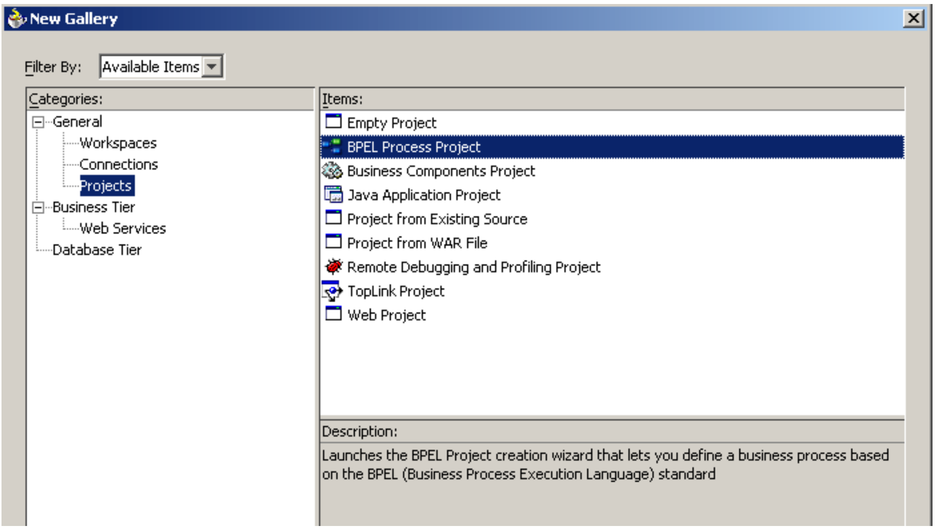
To create a new BPEL project for inbound interaction:

1. Click the **Applications** tab and select a workspace for your project.


Oracle JDeveloper 10g - Workspace
File Edit View Search Navigate Run
Applications - Navigator
Applications
Workspace1

2. Right-click the workspace and select **New Project**.

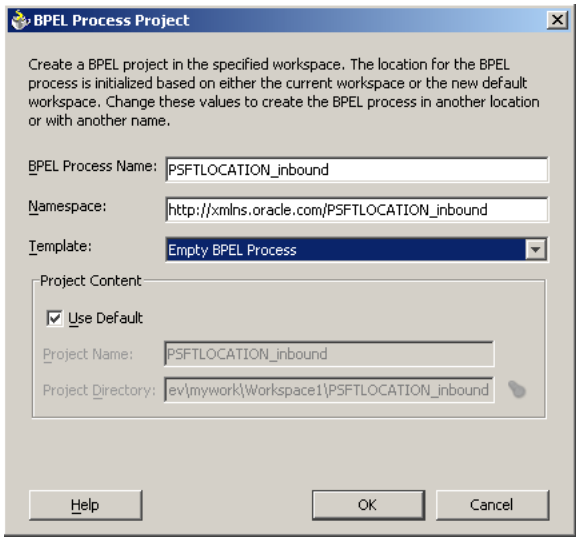
   The New Gallery window is displayed.



Launches the BPEL Project creation wizard that lets you define a business process based on the BPEL (Business Process Execution Language) standard

3. From the Items list, select **BPEL Process Project** and click **OK**.

   The BPEL Process Project dialog box is displayed.



Create a BPEL project in the specified workspace. The location for the BPEL process is initialized based on either the current workspace or the new default workspace. Change these values to create the BPEL process in another location or with another name.

BPEL Process Name: PSFTLOCATION_inbound

Namespace: http://xmlns.oracle.com/PSFTLOCATION_inbound

Template: Empty BPEL Process

☑ Use Default

Project Name: PSFTLOCATION_inbound

Project Directory: ev\mywork\Workspace1\PSFTLOCATION_inbound

4. Perform the following steps:

   a. Specify a name for the process, for example, **PSFTLOCATION_inbound**.

      The Namespace field is updated automatically.

   b. From the Template list, select **Empty BPEL Process**.

   c. Click **OK**.

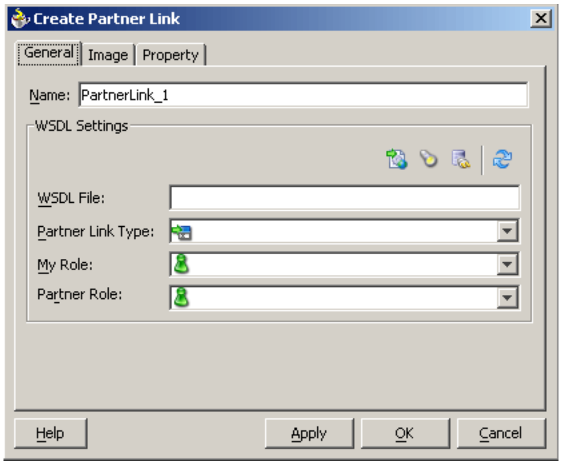
Create an Inbound PartnerLink Activity

When designing a BPEL process, a PartnerLink activity must be created to invoke the PeopleSoft service. A PartnerLink describes a set of operations within a Web service. The WSDL document is the external contract to which the Web service conforms. Given a WSDL, any BPEL process can initiate a Web service through a PartnerLink.

To create an inbound PartnerLink using the WSDL file you generated in Application Explorer:

1. From the Process Activities pane on the right, drag and drop a PartnerLink to the visual editor.

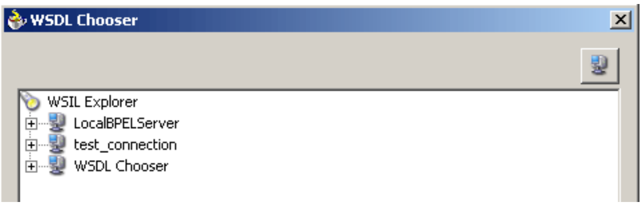
   The Create Partner Link dialog box is displayed.



Name: PartnerLink_1

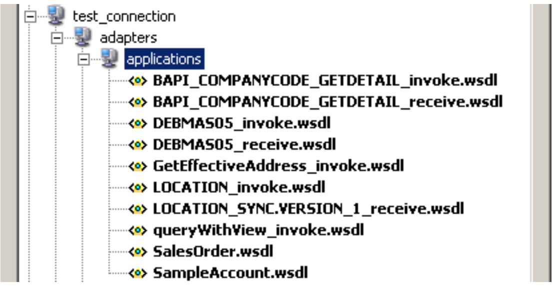
2. Click the **WSIL browser** icon (second icon from the left preceding the **WSDL File** field).

   The WSDL Chooser dialog box is displayed.



3. Expand your new BPEL Server connection, then expand **adapters**, and then **applications**.

The WSDL Chooser dialog box is displayed.



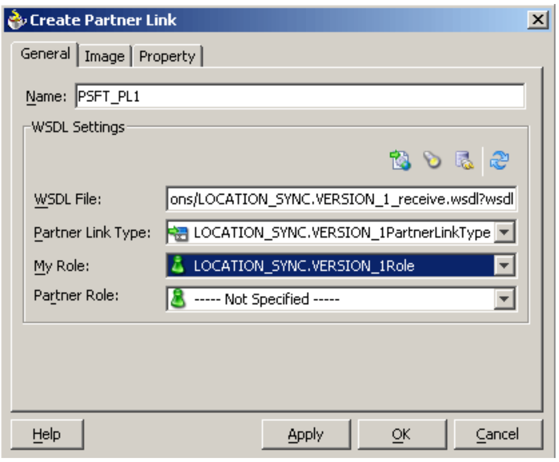
test_connection
adapters
applications
BAPI_COMPANYCODE_GETDETAIL_invoke.wsdl
BAPI_COMPANYCODE_GETDETAIL_receive.wsdl
DEBMAS05_invoke.wsdl
DEBMAS05_receive.wsdl
GetEffectiveAddress_invoke.wsdl
LOCATION_invoke.wsdl
LOCATION_SYNC.VERSION_1_receive.wsdl
queryWithView_invoke.wsdl
SalesOrder.wsdl
SampleAccount.wsdl

Note: If you have organized your WSDL files in subfolders, the WSIL browser will display the full tree structure of your WSDL hierarchy. By default, the names of all WSDL files generated for inbound adapter services end with `_receive`.

---

4. Select **LOCATION_SYNC.VERSION_1_receive.wsdl** and click **OK**.

   The Create Partner Link dialog box is displayed.



Name: PSFT_PL1

WSDL File: ons/LOCATION_SYNC.VERSION_1_receive.wsdl?wsdl

Partner Link Type: LOCATION_SYNC.VERSION_1PartnerLinkType

My Role: LOCATION_SYNC.VERSION_1Role

Partner Role: ----- Not Specified -----

   The **WSDL File** field displays the name and location of the selected WSDL file. The **Partner Link Type** field specifies the PartnerLink defined in the WSDL file.

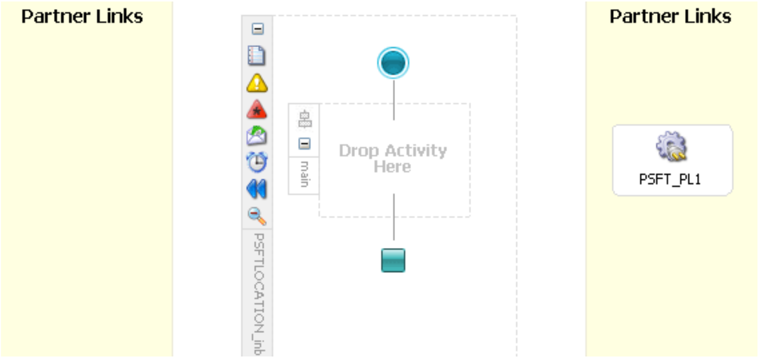
   Perform the following steps:

   a. From the **My Role** list, select the default value **LOCATION_SYNC.VERSION_1Role**.

   b. Leave the **Partner Role** field unspecified.

5. Click **Apply**, and then **OK**.

   The new PSFT_PL1 PartnerLink appears in the visual editor.



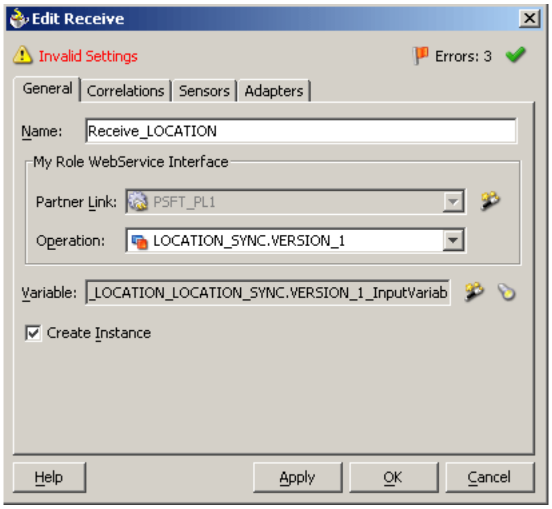
6. Select **Save** from the **File** menu.

Create an Inbound Receive Activity

To create an inbound Receive Activity:

1. From the **Process Activities** pane on the right, drag a **Receive** activity to the visual editor and place it in the designated placeholder labeled **Drop Activity Here**.
2. Extend a connection between the **Receive** activity and the **PartnerLink** you created.

   The Edit Receive dialog box is displayed.



Name: Receive_LOCATION

My Role WebService Interface

Partner Link: PSFT_PL1

Operation: LOCATION_SYNC.VERSION_1

Variable: _LOCATION_LOCATION_SYNC.VERSION_1_InputVariab

☑ Create Instance

   Perform the following steps:

   a. Specify a name for the Receive Activity, for example, **Receive_LOCATION**.
   b. Click the first icon to the right of the **Variable** field, then click **OK** in the Create Variable dialog box that is displayed.
   c. Verify that the **Create Instance** check box is selected.
3. Click **Apply**.

   The Receive dialog box should no longer display any warnings or errors.
4. Click **OK**.

   A connection is created between the PartnerLink and the Receive activity. You have completed the design of your inbound BPEL process.

See "Listening to Adapter Events Inside BPEL Process Manager" on page 4-18 for information on how to deploy and manage your inbound process.

See Also:

- *Oracle BPEL Process Manager Developer's Guide*
- *Oracle Application Server Adapter Concepts*

Invoking Adapter Request-Response Service from BPEL Process Manager

The OracleAS Adapter for PeopleSoft request-response service is used to create, delete, update, and query back-end data as well as to call back-end workflows and transactions. The following section describes how to invoke the adapter synchronous request-response service, also referred to as Outbound Interaction, as well as how to manage the process in Oracle BPEL Console.

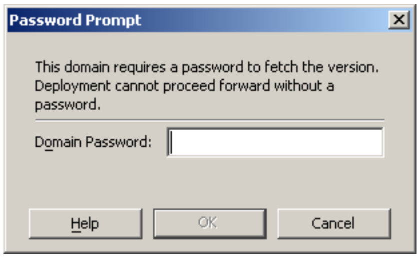
Deploy the Outbound BPEL Process

The procedures for deploying an inbound and an outbound BPEL process using the JDeveloper interface are identical.

To deploy your BPEL process in JDeveloper:

1. Right-click your process flow in the Applications - Navigator pane.
2. Select **Deploy**, then ***Your BPEL PM Server connection***, and then **Deploy to default domain**.

   The Password Prompt dialog box is displayed.



Password Prompt

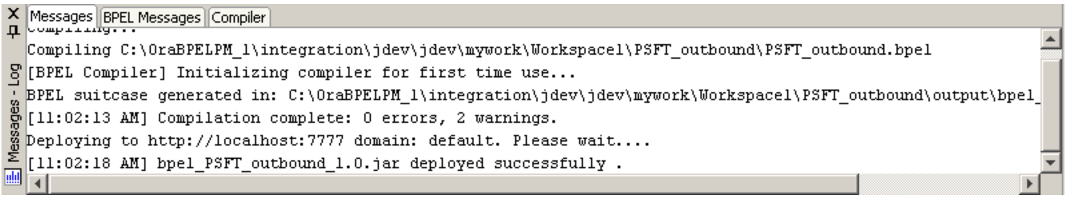
This domain requires a password to fetch the version. Deployment cannot proceed forward without a password.

Domain Password:

3. In the **Domain Password** field, enter your BPEL Process Manager password.

   The deployment process starts automatically after you enter the correct password.
4. Observe the **Messages** log at the bottom of the window.

   The Messages log displays the deployment status. In this example, it shows a successful deployment message for the process.



```
Compiling...
Compiling C:\OraBPELPM_1\integration\jdev\jdev\mywork\Workspace1\PSFT_outbound\PSFT_outbound.bpel
[BPEL Compiler] Initializing compiler for first time use...
BPEL suitcase generated in: C:\OraBPELPM_1\integration\jdev\jdev\mywork\Workspace1\PSFT_outbound\output\bpel_
[11:02:13 AM] Compilation complete: 0 errors, 2 warnings.
Deploying to http://localhost:7777 domain: default. Please wait....
[11:02:18 AM] bpel_PSFT_outbound_1.0.jar deployed successfully .
```

   If deployment was not successful, click the **Compiler** tab to view all error and warning messages generated during the deployment process.

Manage the Deployed Outbound Process in Oracle BPEL Console

JDeveloper deploys the developed process directly to the Oracle BPEL Console, which enables you to run, monitor, and administer BPEL processes.

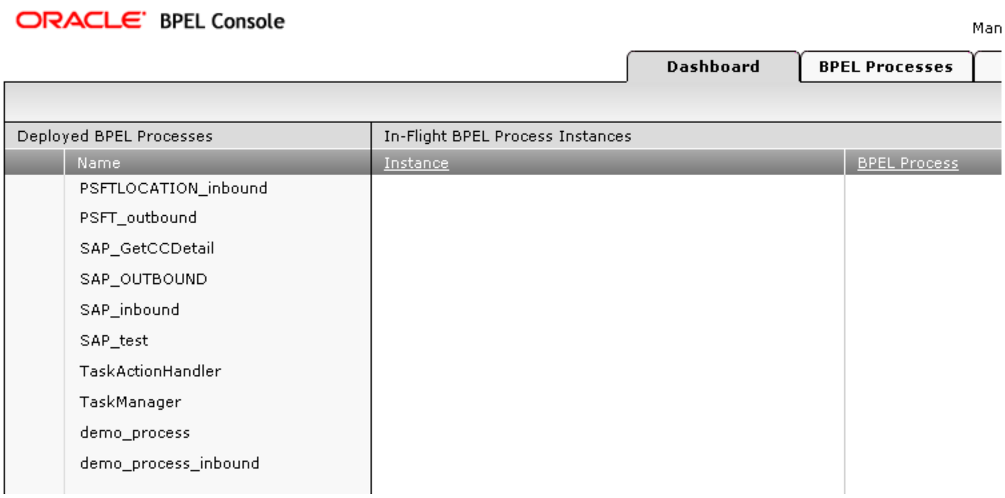
To invoke adapter request-response service:

1. Start the Oracle BPEL Console by entering the following URL in a browser:

   `http://host:port/BPELConsole`

2. Select a domain and provide a valid password.

   The Oracle BPEL Console main page is displayed. All deployed BPEL processes are listed in the Dashboard tab.



Deployed BPEL Processes

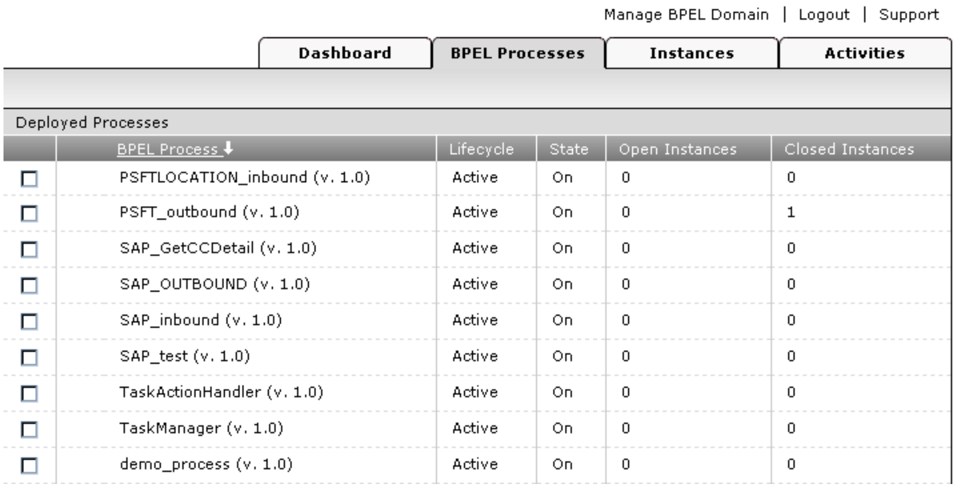
Name
PSFTLOCATION_inbound
PSFT_outbound
SAP_GetCCDetail
SAP_OUTBOUND
SAP_inbound
SAP_test
TaskActionHandler
TaskManager
demo_process
demo_process_inbound

In-Flight BPEL Process Instances

Instance	BPEL Process

3. Click the **BPEL Processes** tab.

   This tab provides a more detailed view of each deployed process.

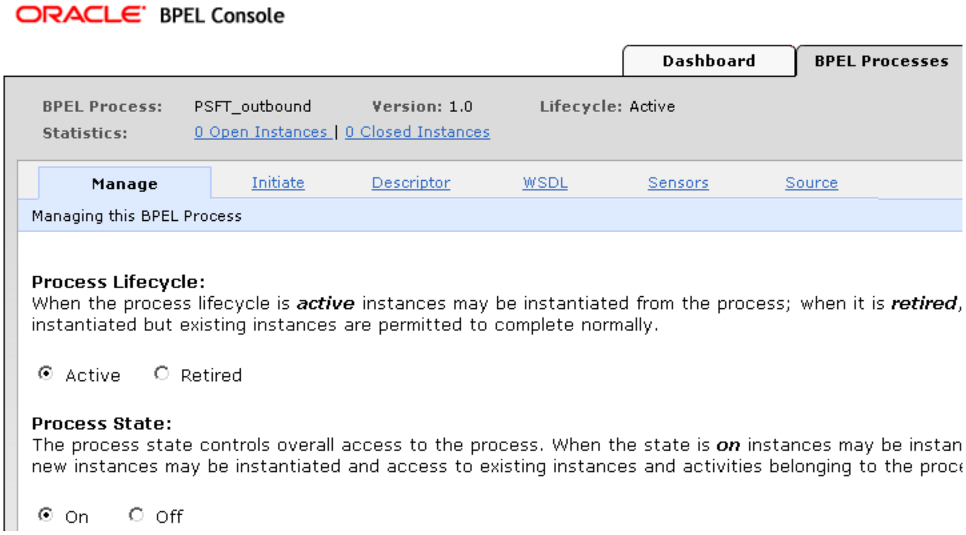


Deployed Processes

	BPEL Process ⬇	Lifecycle	State	Open Instances	Closed Instances
□	PSFTLOCATION_inbound (v. 1.0)	Active	On	0	0
□	PSFT_outbound (v. 1.0)	Active	On	0	1
□	SAP_GetCCDetail (v. 1.0)	Active	On	0	0
□	SAP_OUTBOUND (v. 1.0)	Active	On	0	0
□	SAP_inbound (v. 1.0)	Active	On	0	0
□	SAP_test (v. 1.0)	Active	On	0	0
□	TaskActionHandler (v. 1.0)	Active	On	0	0
□	TaskManager (v. 1.0)	Active	On	0	0
□	demo_process (v. 1.0)	Active	On	0	0

4. Click the PeopleSoft process link, `PSFT_outbound (v. 1.0)`.

   The Manage window provides options for managing this BPEL process. Do not change any of the following default settings.



BPEL Process: PSFT_outbound Version: 1.0 Lifecycle: Active
Statistics: 0 Open Instances | 0 Closed Instances

Managing this BPEL Process

Process Lifecycle:
When the process lifecycle is ***active*** instances may be instantiated from the process; when it is ***retired***, instantiated but existing instances are permitted to complete normally.

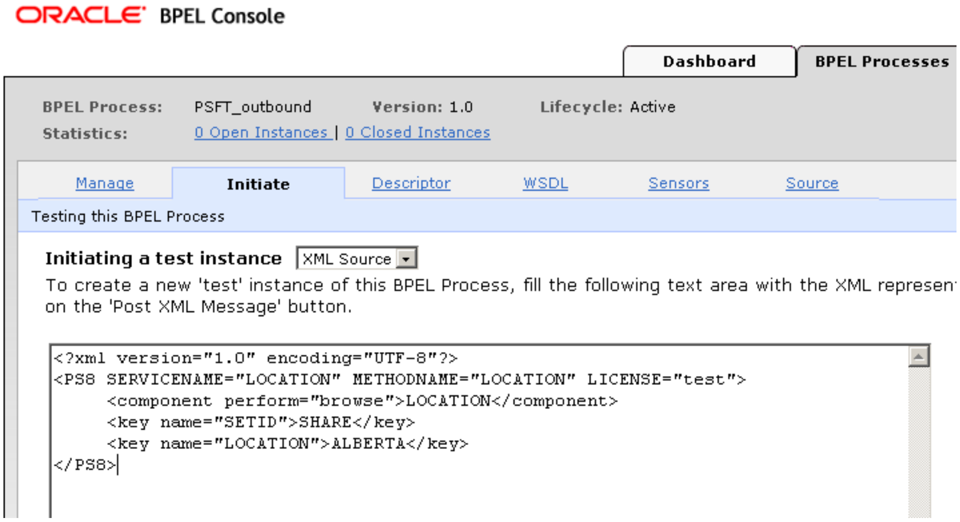
● Active ○ Retired

Process State:
The process state controls overall access to the process. When the state is ***on*** instances may be instan new instances may be instantiated and access to existing instances and activities belonging to the proce

● On ○ Off

5. Click the **Initiate** tab.

   The Initiate tab enables you to test your BPEL process.



BPEL Process: PSFT_outbound Version: 1.0 Lifecycle: Active
Statistics: 0 Open Instances | 0 Closed Instances

Testing this BPEL Process

Initiating a test instance XML Source

To create a new 'test' instance of this BPEL Process, fill the following text area with the XML represen on the 'Post XML Message' button.

```
<?xml version="1.0" encoding="UTF-8"?>
<PS8 SERVICENAME="LOCATION" METHODNAME="LOCATION" LICENSE="test">
     <component perform="browse">LOCATION</component>
     <key name="SETID">SHARE</key>
     <key name="LOCATION">ALBERTA</key>
</PS8>
```

   Perform the following steps:

   a. From the **Initiating a test instance** list, select **XML Source**.

   b. Enter the following code in the text area provided for XML input:

```
<?xml version="1.0" encoding="UTF-8"?>
<PS8 SERVICENAME="LOCATION" METHODNAME="LOCATION" LICENSE="test">
     <component perform="browse">LOCATION</component>
     <key name="SETID">SHARE</key>
     <key name="LOCATION">ALBERTA</key>
</PS8>
```

6. Click **Post XML Message**.

   The response received from the PeopleSoft system is displayed in the Initiate window.

   See Also: *Oracle Application Server Adapter Concepts*

Listening to Adapter Events Inside BPEL Process Manager

The OracleAS Adapter for PeopleSoft event notification service, also referred to as Inbound Interaction, is used to listen to events that occur in an EIS. The following section describes how to deploy your inbound BPEL process and listen to adapter events at runtime using Oracle BPEL Console.

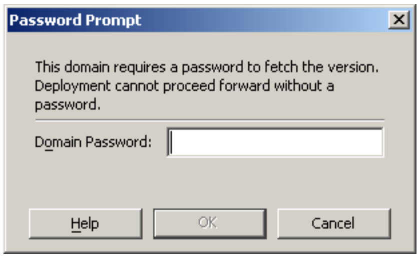
Deploy the Inbound BPEL Process

The procedures for deploying an inbound and an outbound BPEL process using the JDeveloper interface are identical.

To deploy your BPEL process in JDeveloper:

1. Right-click your process flow in the Applications pane.
2. Select **Deploy**, then ***Your BPEL PM Server connection***, and then **Deploy to default domain**.

   The Password Prompt dialog box is displayed.



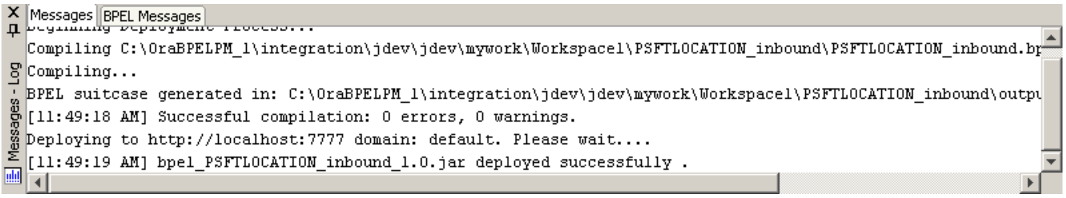
This domain requires a password to fetch the version. Deployment cannot proceed forward without a password.

Domain Password:

3. In the **Domain Password** field, enter your BPEL Process Manager password.

   The deployment process starts automatically after you enter the correct password.
4. Observe the **Messages** log at the bottom of the window.

   The Messages log displays the deployment status. In this example, it shows a successful deployment message for the process.



```
Beginning Deployment Process...
Compiling C:\OraBPELPM_1\integration\jdev\jdev\mywork\Workspace1\PSFTLOCATION_inbound\PSFTLOCATION_inbound.b
Compiling...
BPEL suitcase generated in: C:\OraBPELPM_1\integration\jdev\jdev\mywork\Workspace1\PSFTLOCATION_inbound\outpu
[11:49:18 AM] Successful compilation: 0 errors, 0 warnings.
Deploying to http://localhost:7777 domain: default. Please wait....
[11:49:19 AM] bpel_PSFTLOCATION_inbound_1.0.jar deployed successfully .
```

   If deployment was not successful, click the **Compiler** tab to view all error and warning messages generated during the deployment process.

Listen to Adapter Events in Oracle BPEL Console

JDeveloper deploys the developed process directly to Oracle BPEL Console, which enables you to run, monitor, and administer BPEL processes, as well as to listen to adapter events at runtime using Oracle BPEL Console.

To listen to adapter events:

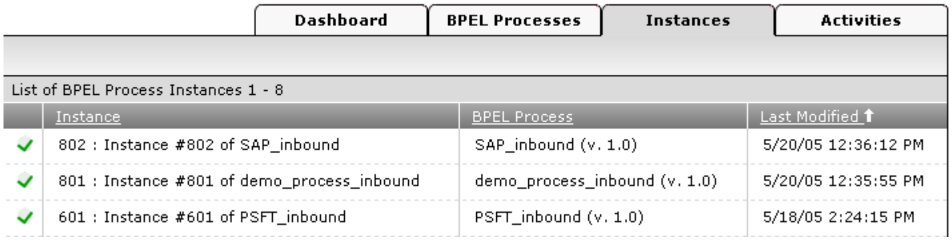
1. Start the Oracle BPEL Console by entering the following URL in a browser:

   `http://host:port/BPELConsole`
2. Select a domain and provide a valid password.

The Oracle BPEL Console Dashboard tab is displayed.

3. Click the **Instances** tab.

   Upon receiving a runtime event, an instance of the event is displayed under the Instances tab.



List of BPEL Process Instances 1 - 8

	Instance	BPEL Process	Last Modified ↑
✓	802 : Instance #802 of SAP_inbound	SAP_inbound (v. 1.0)	5/20/05 12:36:12 PM
✓	801 : Instance #801 of demo_process_inbound	demo_process_inbound (v. 1.0)	5/20/05 12:35:55 PM
✓	601 : Instance #601 of PSFT_inbound	PSFT_inbound (v. 1.0)	5/18/05 2:24:15 PM

4. To see the event message, click the instance, and then clickthe **Audit** tab.

   The event message is displayed.

5. Click **More...** to view the entire message, or **View Raw XML** to view the XML source.

See Chapter 5, "BPEL Process Manager Integration Examples" for more information.

See Also: *Oracle Application Server Adapter Concepts*

5

BPEL Process Manager Integration Examples

This chapter contains the following examples:

- PeopleSoft Service Integration
- PeopleSoft Event Integration

The scenarios shown in this chapter require the following prerequisites.

Prerequisites

The following are installation and configuration requirements:

- OracleAS Adapter for PeopleSoft must be installed on Oracle Application Server.
- Oracle BPEL PM Server must be properly configured and running.
- Oracle JDeveloper must be properly installed.

> **See Also:** *Oracle Application Server Adapters Installation Guide*

The examples in this chapter present the configuration steps necessary for demonstrating service and event integration with PeopleSoft. Prior to using this material, you must be familiar with the following:

- How to create a J2CA configuration, as BPEL PM is only compatible with the J2CA Connector. See "Creating a Configuration for J2CA" on page 2-8 for more information.
- How to configure OracleAS Adapter for PeopleSoft for services and events using Application Explorer. See Chapter 2, "Configuring OracleAS Adapter for PeopleSoft" for more information.

> **See Also:** *Oracle BPEL Process Manager Developer's Guide*

Adapter integration with Oracle BPEL Process Manager is a two-step process:

1. **Design Time:** OracleAS Adapter for PeopleSoft is configured in Application Explorer for services and events, as described in Chapter 2, "Configuring OracleAS Adapter for PeopleSoft". Integration logic is modeled using JDeveloper.
2. **Runtime:** After you deploy the BPEL process you designed in JDeveloper, you can test your service configuration or see newly received events in the BPEL Console.

PeopleSoft Service Integration

This topic illustrates PeopleSoft service integration. Design-time and runtime configuration is described.

Design-Time Configuration

Before you design a process for PeopleSoft service integration, you must generate its respective WSDL file using Application Explorer.

Generating WSDL for a Request-Response J2CA Service

To generate WSDL for outbound interaction in Application Explorer:

1. Start **Application Explorer** and connect to a defined PeopleSoft target or create a new target.

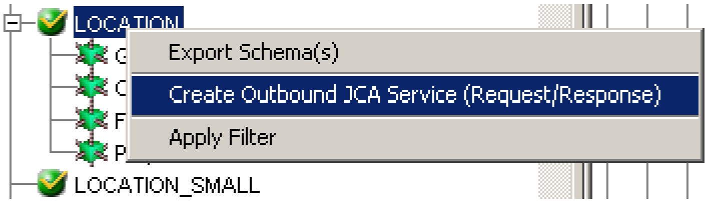
   See "Defining a Target to PeopleSoft" on page 2-10 for more information.

2. Expand the PeopleSoft target to which you are connected.

3. Expand **Component Interfaces** and select **LOCATION**.

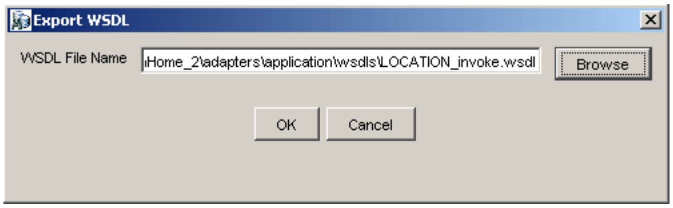
4. Right-click **LOCATION**.

   The following menu is displayed:



5. Click **Create Outbound JCA Service (Request/Response)**.

   The Export WSDL dialog box is displayed.



WSDL File Name Home_2\adapters\application\wsdls\LOCATION_invoke.wsdl

6. Click **OK**.

You are ready to design a BPEL process in JDeveloper.

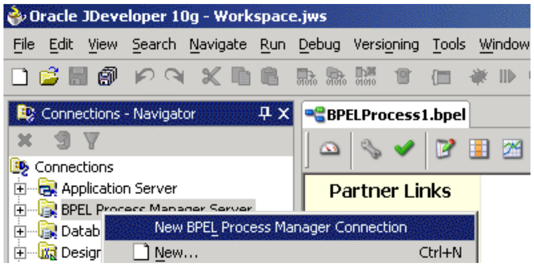
Creating a BPEL PM Server Connection

Before you design a BPEL process, you must create a connection to your BPEL Server using JDeveloper. To create a server connection:

1. Open **JDeveloper**.

2. To display the connections, click the **Connections** tab at the bottom of the upper left pane in JDeveloper.

   The following menu is displayed.

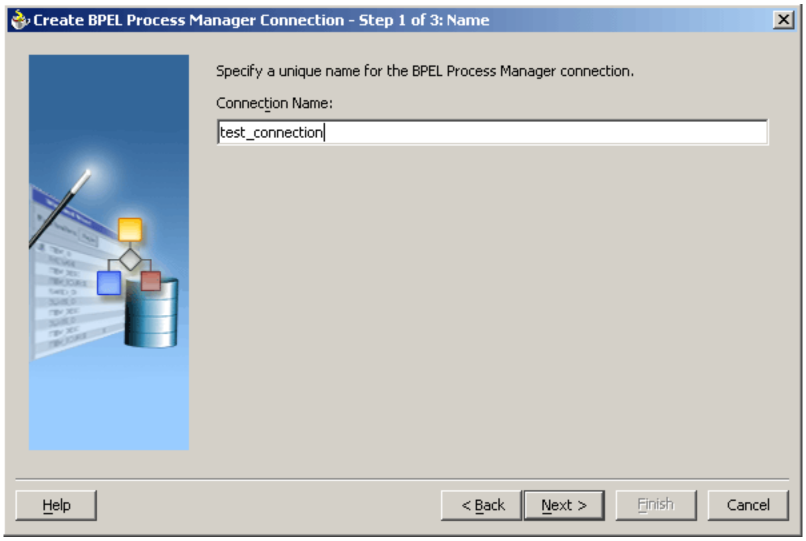

Oracle JDeveloper 10g - Workspace.jws
File Edit View Search Navigate Run Debug Versioning Tools Window
Connections - Navigator
BPELProcess1.bpel
Connections
Application Server
BPEL Process Manager Server
Datab
Design
Partner Links
New BPEL Process Manager Connection
New... Ctrl+N

3. Right-click **BPEL Process Manager Server** and select **New BPEL Process Manager Connection**.

   The Create BPEL Process Manager Connection - Welcome dialog box is displayed.

4. Click **Next**.

   The Create BPEL Process Manager Connection dialog box is displayed.


Create BPEL Process Manager Connection - Step 1 of 3: Name

Specify a unique name for the BPEL Process Manager connection.

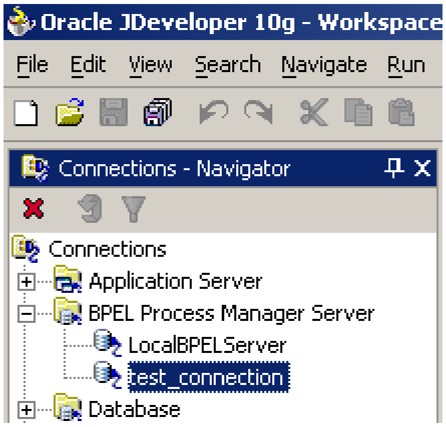
Connection Name: test_connection

5. Specify a unique name for your BPEL Server connection and click **Next**.

6. Specify a valid host name and port number for the BPEL PM Server you wish to connect to.

7. Click **Finish**.

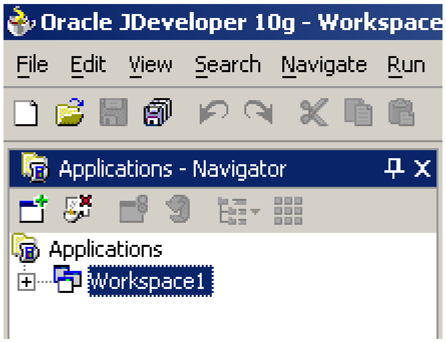
   Your newly created server connection is displayed in the Connections tab under the BPEL Process Manager Server node.


Oracle JDeveloper 10g - Workspace
File Edit View Search Navigate Run
Connections - Navigator
Connections
Application Server
BPEL Process Manager Server
LocalBPELServer
test_connection
Database

Creating a BPEL Project for a Synchronous BPEL Process

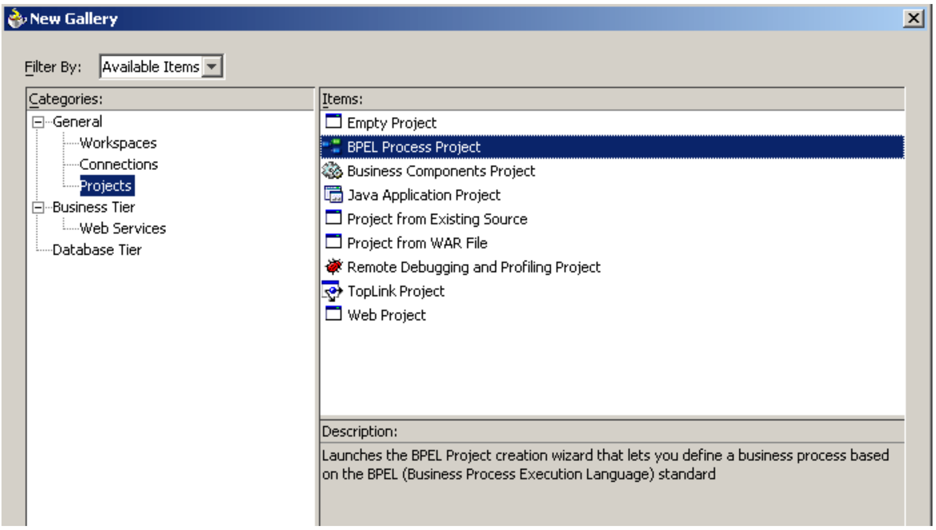
To create a BPEL Project for a synchronous BPEL process:

1. At the bottom of the upper left pane, click the **Applications** tab and select a workspace for your project.


Oracle JDeveloper 10g - Workspace
File Edit View Search Navigate Run
Applications - Navigator
Applications
Workspace1

2. Right-click the workspace and select **New Project**.

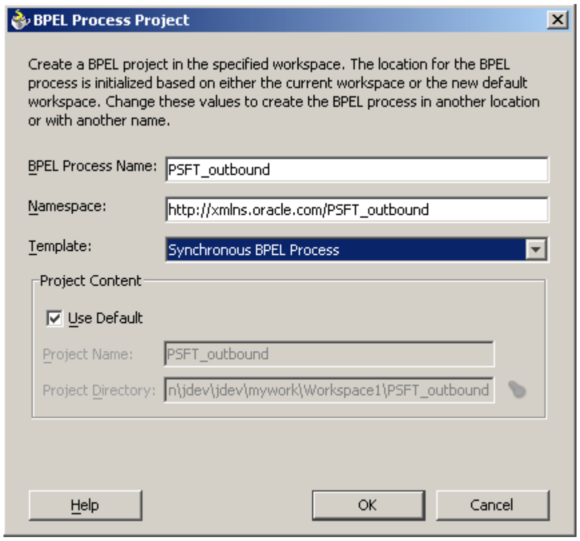
   The New Gallery window is displayed.



Launches the BPEL Project creation wizard that lets you define a business process based on the BPEL (Business Process Execution Language) standard

3. From the Items list, select **BPEL Process Project** and click **OK**.

   The BPEL Process Project dialog box is displayed.



BPEL Process Project

Create a BPEL project in the specified workspace. The location for the BPEL process is initialized based on either the current workspace or the new default workspace. Change these values to create the BPEL process in another location or with another name.

BPEL Process Name: PSFT_outbound

Namespace: http://xmlns.oracle.com/PSFT_outbound

Template: Synchronous BPEL Process

Project Content

☑ Use Default

Project Name: PSFT_outbound

Project Directory: n\jdev\jdev\mywork\Workspace1\PSFT_outbound

4. Perform the following steps:

   a. Specify a name for the BPEL process.

      The Namespace field is updated automatically.

   b. From the Template list, select **Synchronous BPEL Process**.

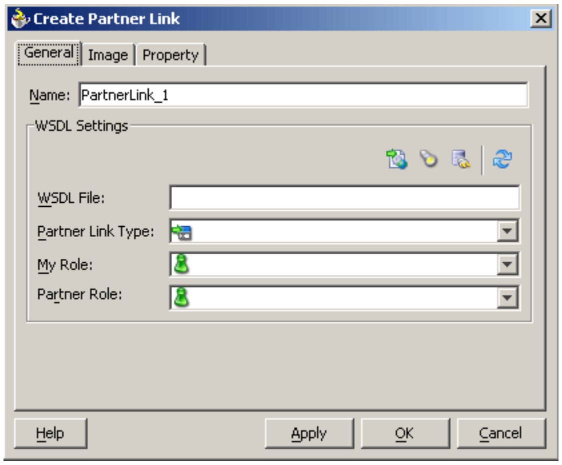
5. Click **OK**.

Designing the BPEL Process for the LOCATION Outbound Service

To design the BPEL Process:

1. From the Process Activities pane on the right, drag and drop a PartnerLink to the visual editor.

   The Create Partner Link dialog box is displayed.



Create Partner Link

General | Image | Property

Name: PartnerLink_1

WSDL Settings

WSDL File:

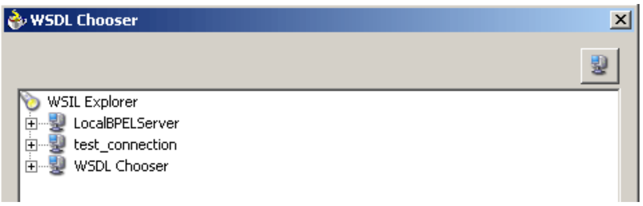
Partner Link Type:

My Role:

Partner Role:

2. Click the **WSIL browser** icon (second icon from the left preceding the **WSDL File** field).

The WSDL Chooser dialog box is displayed.

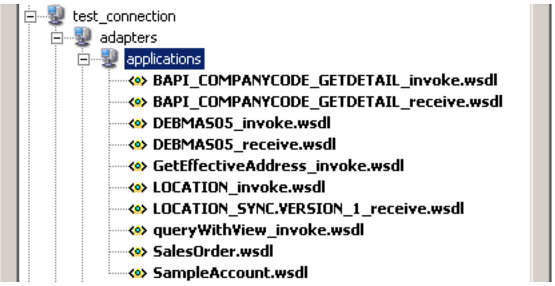


WSDL Chooser

WSIL Explorer
- LocalBPELServer
- test_connection
- WSDL Chooser

3. Expand your new server connection, then expand **adapters**, and then **applications**.

   The WSDL tree is displayed.



test_connection
- adapters
  - applications
    - BAPI_COMPANYCODE_GETDETAIL_invoke.wsdl
    - BAPI_COMPANYCODE_GETDETAIL_receive.wsdl
    - DEBMAS05_invoke.wsdl
    - DEBMAS05_receive.wsdl
    - GetEffectiveAddress_invoke.wsdl
    - LOCATION_invoke.wsdl
    - LOCATION_SYNC.VERSION_1_receive.wsdl
    - queryWithView_invoke.wsdl
    - SalesOrder.wsdl
    - SampleAccount.wsdl

4. Select **LOCATION_invoke.wsdl** and click **OK**.

   The **WSDL File** field in the Create Partner Link dialog box displays the name and location of the selected WSDL file. The **Partner Link Type** field specifies the PartnerLink defined in the WSDL file.

   Perform the following steps:

   a. Leave the **My Role** field unspecified. The role of the PartnerLink is null, as it will be synchronously invoked from the BPEL process.

   b. From the **Partner Role** list, select the default value **LOCATIONRole**. This is the role of the BPEL process.

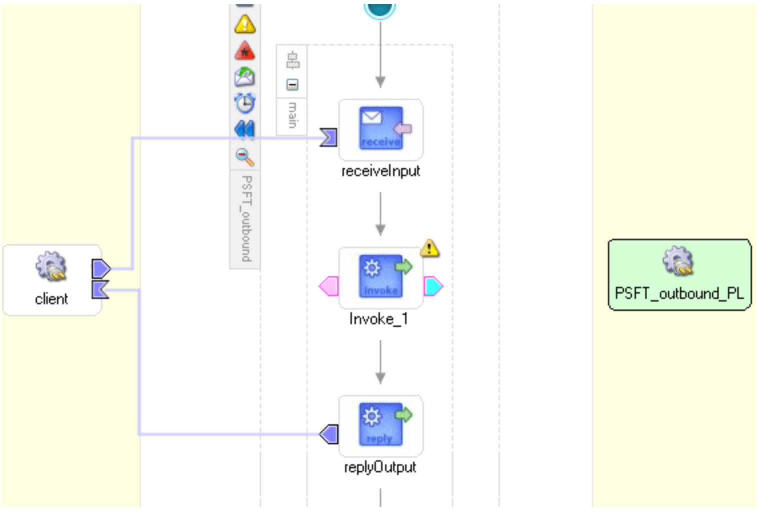
5. Click **OK**.

   The new PartnerLink appears in the visual editor.

6. Select **Save** from the File menu.

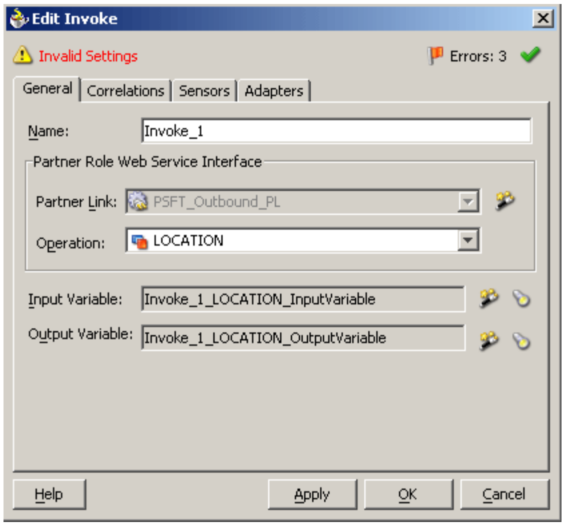
7. From the **Process Activities** pane on the right, drag an **Invoke** activity to the visual editor and place it between the Receive activity (`receiveInput`) and the Reply activity (`replyOutput`).

The Invoke process activity is shown in the following diagram view.



8. Drag the **blue arrow** from Invoke_1 and connect it to the PeopleSoft PartnerLink.

   The Edit Invoke dialog box is displayed.



Edit Invoke

Invalid Settings   Errors: 3

Name: Invoke_1

Partner Role Web Service Interface

Partner Link: PSFT_Outbound_PL

Operation: LOCATION

Input Variable: Invoke_1_LOCATION_InputVariable

Output Variable: Invoke_1_LOCATION_OutputVariable

Perform the following steps:

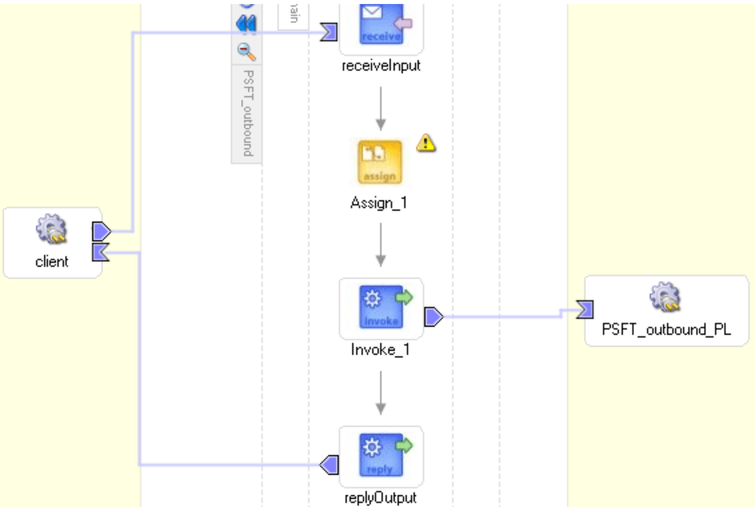
a. Click the first icon to the right of the **Input Variable** field, then click **OK** in the Create Variable window that is displayed.

b. Repeat the previous step to create a default variable for Output Variable.

9. Click **OK**.

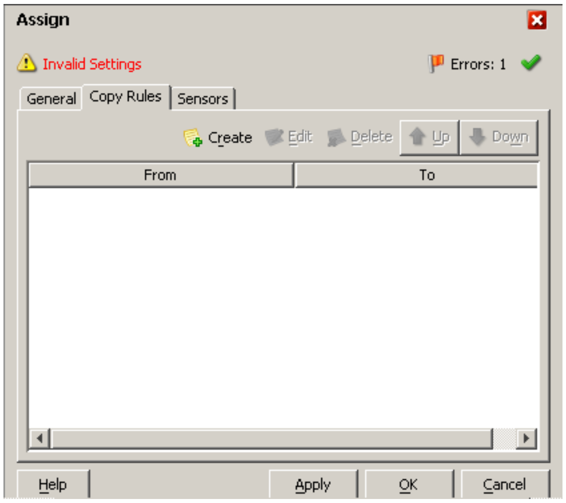
10. Drag an **Assign** process activity and drop it between `receiveInput` and `Invoke_1`.

The following image shows the new Assign activity in JDeveloper visual editor.



11. Double-click the **Assign** activity icon.

    The Assign dialog box is displayed.



12. In the Copy Rules tab, click **Create**.

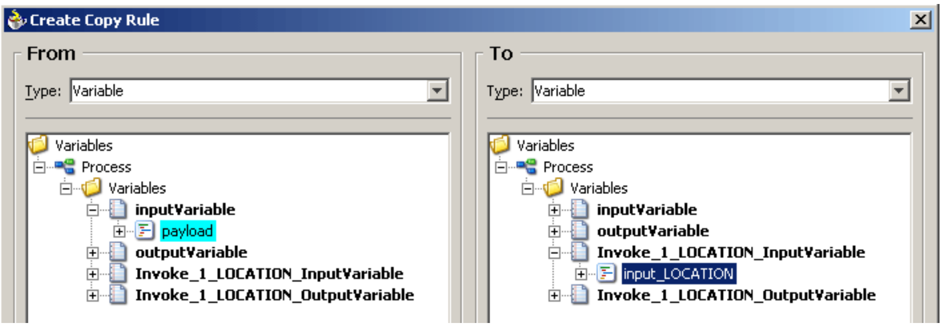
    The Create Copy Rule dialog box is displayed.

    Perform the following steps:

    a. In the **From** pane, expand **Variables**, then **inputVariable**, and then highlight **payload**.

    b. In the **To** pane, expand **Variables**, then **Invoke_1_LOCATION_InputVariable**, and then highlight **input_LOCATION**.

Your Create Copy Rule dialog box should look as follows:



From Type: Variable

To Type: Variable

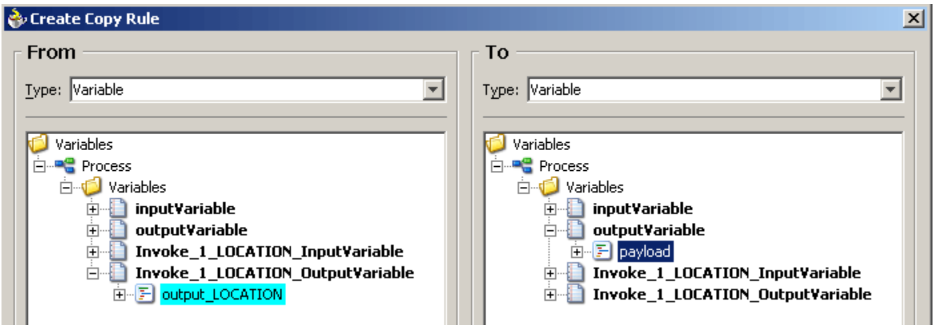
13. To close the Create Copy Rule dialog box and the Assign dialog box, click **OK**.

14. From the **Process Activities** pane on the right, drag another **Assign** activity to the visual editor and place it between the Invoke activity (`Invoke_1`) and the Reply activity (`replyOutput`).

15. Double-click the **Assign** activity icon and click **Create**.

16. In the Create Copy Rule dialog box, map **Invoke_1_LOCATION_OutputVariable, output_LOCATION** to **outputVariable, payload**.

    Verify that you have mapped all variables as follows:



From Type: Variable

To Type: Variable

17. Click **OK**, then click **OK** again.

18. Select **Save** from the File menu.

You have completed the design of your BPEL process.

Deploying the BPEL Process for the LOCATION Outbound Service

JDeveloper deploys the outbound BPEL process directly to Oracle BPEL Console.

To deploy your BPEL process in JDeveloper:

1. Right-click your process flow in the Applications - Navigator pane.

2. Select **Deploy**, then ***Your BPEL PM Server connection***, and then **Deploy to default domain**.

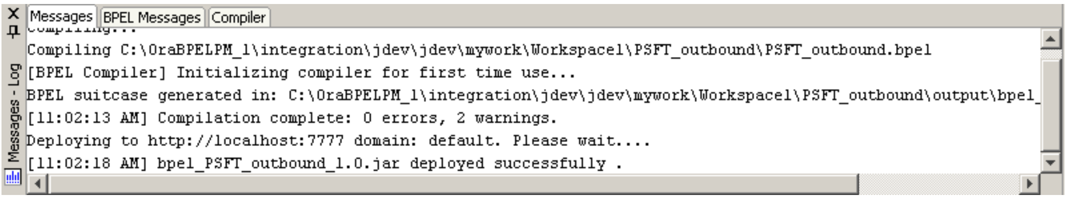
   The Password Prompt dialog box is displayed.

3. Enter your BPEL Process Manager password in the Password Prompt dialog box.

   The deployment process starts automatically after you enter the correct password.

4. Observe the **Messages** log at the bottom of the window.

The Messages log displays the deployment status. In this example, it shows a successful deployment message for the process.



```
Compiling...
Compiling C:\OraBPELPM_1\integration\jdev\jdev\mywork\Workspace1\PSFT_outbound\PSFT_outbound.bpel
[BPEL Compiler] Initializing compiler for first time use...
BPEL suitcase generated in: C:\OraBPELPM_1\integration\jdev\jdev\mywork\Workspace1\PSFT_outbound\output\bpel_
[11:02:13 AM] Compilation complete: 0 errors, 2 warnings.
Deploying to http://localhost:7777 domain: default. Please wait....
[11:02:18 AM] bpel_PSFT_outbound_1.0.jar deployed successfully .
```

If deployment was not successful, click the **Compiler** tab to view all error and warning messages generated during the deployment process.

Runtime Configuration

To invoke the LOCATION process from Oracle BPEL Console:

1. Start the Oracle BPEL Console by entering the following URL in a browser:

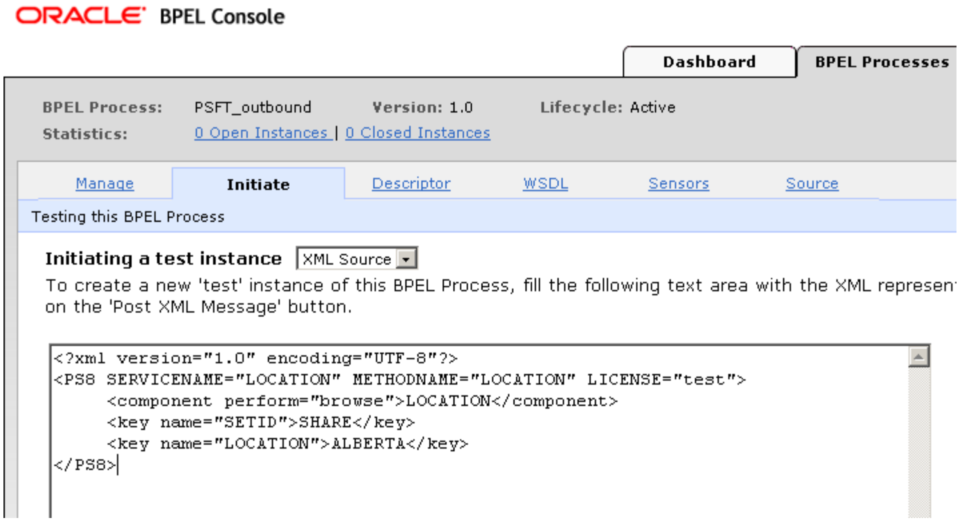
   `http://`*`host`*`:`*`port`*`/BPELConsole`

2. Select a domain and provide a valid password.

   The Oracle BPEL Console main page is displayed.

3. Click the **BPEL Processes** tab.
4. Click the PeopleSoft process link, **PSFT_outbound**.
5. Click **Initiate**.

   The Initiate tab enables you to test your BPEL process.



BPEL Process: PSFT_outbound Version: 1.0 Lifecycle: Active

Statistics: 0 Open Instances | 0 Closed Instances

Testing this BPEL Process

Initiating a test instance

To create a new 'test' instance of this BPEL Process, fill the following text area with the XML represen
on the 'Post XML Message' button.

```
<?xml version="1.0" encoding="UTF-8"?>
<PS8 SERVICENAME="LOCATION" METHODNAME="LOCATION" LICENSE="test">
     <component perform="browse">LOCATION</component>
     <key name="SETID">SHARE</key>
     <key name="LOCATION">ALBERTA</key>
</PS8>
```

Perform the following steps:

a. From the **Initiating a test instance** list, select **XML Source**.

b. Enter the following code in the text area provided for XML input:

```
<?xml version="1.0" encoding="UTF-8"?>
<PS8 SERVICENAME="LOCATION" METHODNAME="LOCATION" LICENSE="test">
     <component perform="browse">LOCATION</component>
     <key name="SETID">SHARE</key>
     <key name="LOCATION">ALBERTA</key>
```

```
</PS8>
```

6. Click **Post XML Message**.

   The response received from the PeopleSoft system is displayed in the Initiate window.

PeopleSoft Event Integration

This topic illustrates PeopleSoft event integration. Design-time and runtime configuration is described.

Design-Time Configuration

You must create a separate channel for every event and select that channel when you generate WSDL for inbound interaction using Application Explorer.

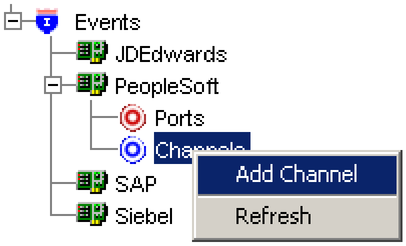
> **Note:** If two or more events share the same channel, event messages may not be delivered to the right BPEL process.

Creating a Channel

To create a channel:

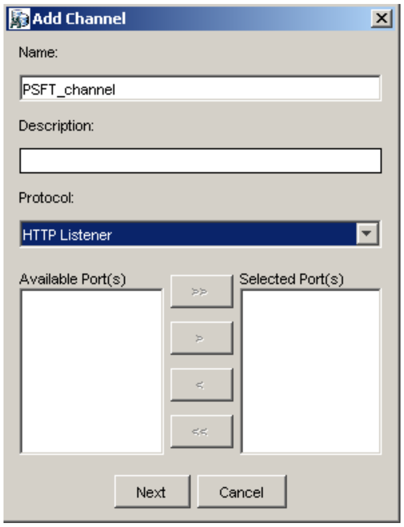
1. Click the **Events** node.
2. Expand the **PeopleSoft** node.

   The ports and channels nodes appear in the left pane.



3. Right-click **Channels** and select **Add Channel**.

The Add Channel dialog box is displayed.



Add Channel

Name: PSFT_channel

Description:

Protocol: HTTP Listener

Provide the following information:

a. Enter a name for the channel, for example, **PSFT_Channel**.

b. Enter a brief description (optional).

c. From the Protocol list, select **HTTP Listener**.

d. **If you are using the optional port feature**, you must select the port you created in Application Explorer. See "J2CA Filtering Port Option for Integration with BPEL Process Manager" on page 2-20 for more information.

   If you are not using event ports for schema validation, skip this step.

4. Click **Next**.

5. When the Basic dialog box is displayed, enter the system information as specified in the following table:

Parameter	Description
Listener port	Port on which to listen for PeopleSoft event data.
Https	For a secure HTTP connection, select the **Https** check box.
Synchronization Type	Choose from the following synchronization options: ■ REQUEST_RESPONSE ■ REQUEST_ACK **Important:** The PeopleSoft channel will not work if the synchronization type is set to REQUEST.

6. Click **OK**.

   The channel appears under the channels node in the left pane. An X over the icon indicates that the channel is currently disconnected.

Note: Do not start the channel, as it is managed by BPEL PM Server. If you start the channel for testing and debugging purposes, stop it before runtime.

Generating WSDL for Event Notification

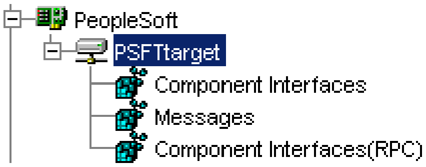
After you create a channel and verify that it is not started, you must generate WSDL for the event using Application Explorer.

1. Start Application Explorer.
2. Expand the **Adapters** node.

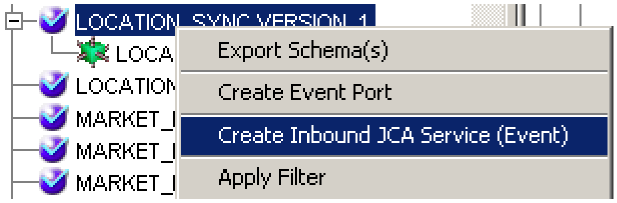
   A list of all available adapters is displayed.
3. Expand **PeopleSoft**.
4. Click a target name under the **PeopleSoft** node, for example, **PSFTtarget**.

   The Logon pane on the right displays the saved parameters.
5. Verify your connection parameters and provide the required password.
6. Right-click the target name and select **Connect**.

   The x icon disappears, indicating that the target is connected.

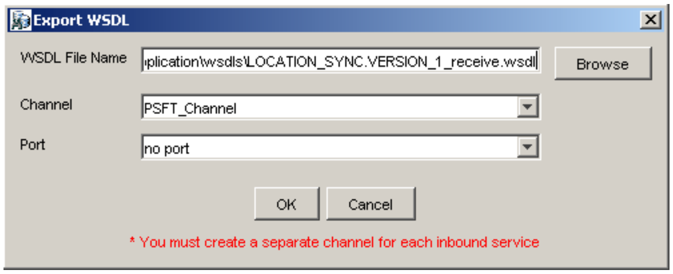


7. Expand **Messages** and select **LOCATION_SYNC.VERSION_1**.
8. Right-click **LOCATION_SYNC.VERSION_1**.



9. Select **Create Inbound JCA Service (Event)**.

   The Export WSDL dialog box is displayed.



WSDL File Name: plication\wsdls\LOCATION_SYNC.VERSION_1_receive.wsdl

Channel: PSFT_Channel

Port: no port

* You must create a separate channel for each inbound service

   Perform the following steps:

   a. In the **WSDL File Name** field, specify the name and location of the WSDL file.

   b. From the **Channel** list, select the channel you created for this inbound service.

      Important: You must create a separate channel for every event. Verify that the channel is stopped before runtime.

c. **If you are using the optional port feature**, you must select a port from the **Port** list. See "J2CA Filtering Port Option for Integration with BPEL Process Manager" on page 2-20 for more information.

   If you are not using event ports for schema validation, skip this step. In this case, the default value of **no port** is selected automatically.

10. Click **OK**.

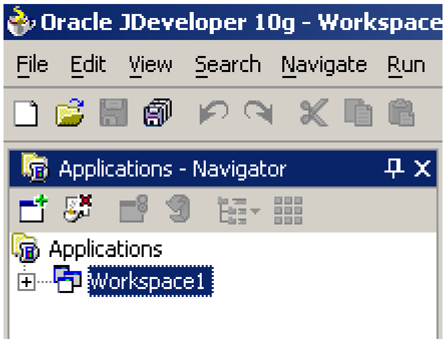
Creating a BPEL PM Server Connection

Before you design a BPEL process, you must create a connection to your BPEL Server using JDeveloper. See "Creating a BPEL PM Server Connection" on page 5-2 for details on how to create the connection.

Designing the BPEL Process for the LOCATION_SYNC.VERSION_1 Inbound Service

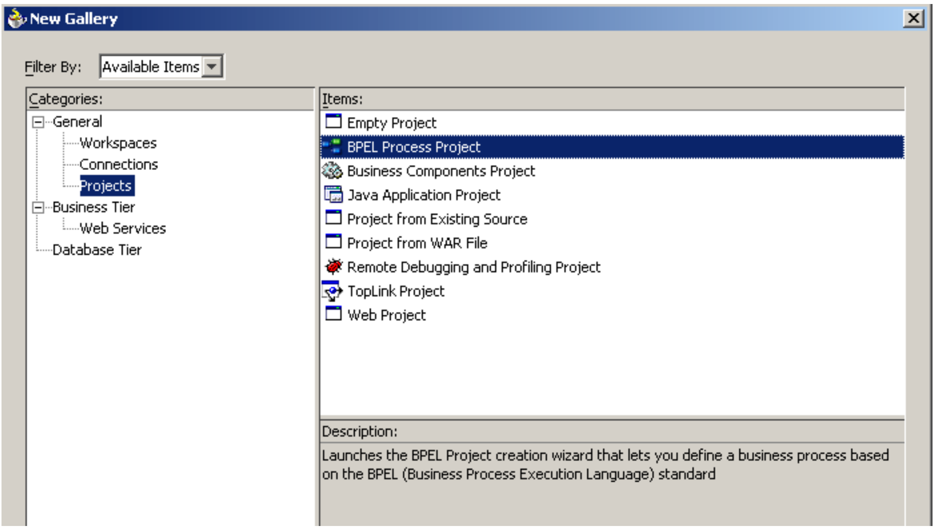
To design a BPEL process for inbound interaction:

1. Click the **Applications** tab and select a workspace for your project.



2. Right-click the workspace and select **New Project**.

   The New Gallery dialog box is displayed.

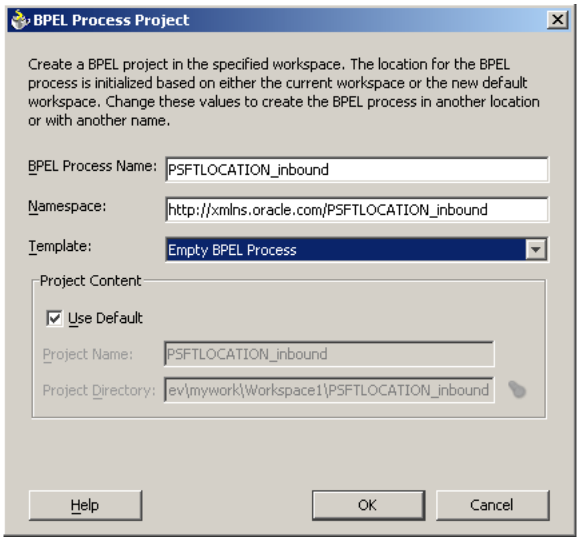


Filter By: Available Items

Description:
Launches the BPEL Project creation wizard that lets you define a business process based on the BPEL (Business Process Execution Language) standard

3. From the Items list, select **BPEL Process Project** and click **OK**.

The BPEL Process Project dialog box is displayed.



BPEL Process Project

Create a BPEL project in the specified workspace. The location for the BPEL process is initialized based on either the current workspace or the new default workspace. Change these values to create the BPEL process in another location or with another name.

BPEL Process Name: PSFTLOCATION_inbound

Namespace: http://xmlns.oracle.com/PSFTLOCATION_inbound

Template: Empty BPEL Process

Project Content

☑ Use Default

Project Name: PSFTLOCATION_inbound

Project Directory: ev\mywork\Workspace1\PSFTLOCATION_inbound

4. Perform the following steps:

   a. Specify a name for the process, for example, **PSFTLOCATION_inbound**.

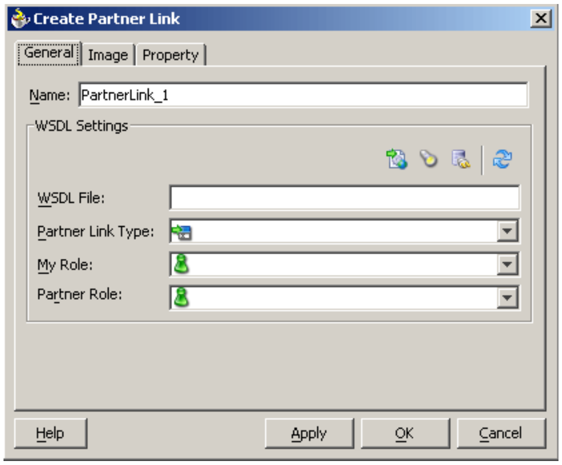
      The Namespace field is updated automatically.

   b. From the Template list, select **Empty BPEL Process**.

   c. Click **OK**.

5. From the Process Activities pane on the right, drag and drop a **PartnerLink** to the visual editor.

   The Create Partner Link dialog box is displayed.



Create Partner Link

Name: PartnerLink_1

WSDL Settings

WSDL File:

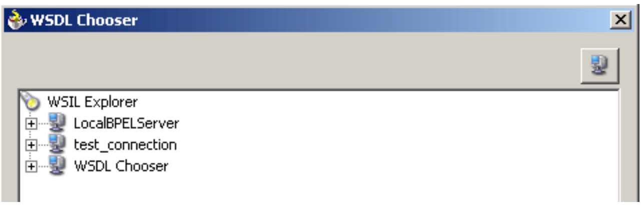
Partner Link Type:

My Role:

Partner Role:

6. Click the **WSIL browser** icon (second icon from the left preceding the **WSDL File** field).

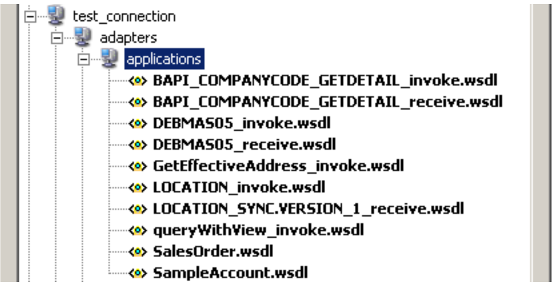
   The WSDL Chooser dialog box is displayed.



WSIL Explorer
- LocalBPELServer
- test_connection
- WSDL Chooser

7. Expand your BPEL Server connection, then expand **adapters**, and then **applications**.

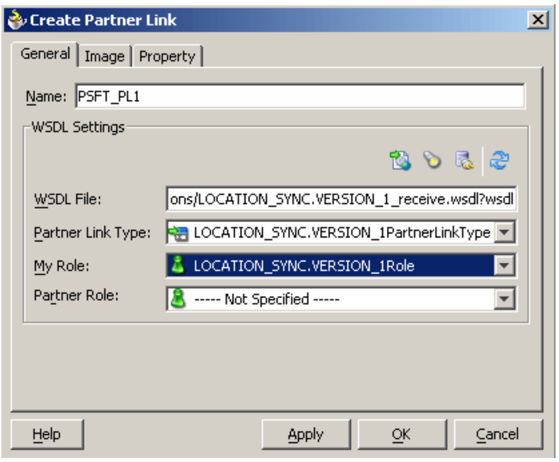
   The WSDL Chooser dialog box is displayed.



test_connection
- adapters
  - applications
    - BAPI_COMPANYCODE_GETDETAIL_invoke.wsdl
    - BAPI_COMPANYCODE_GETDETAIL_receive.wsdl
    - DEBMAS05_invoke.wsdl
    - DEBMAS05_receive.wsdl
    - GetEffectiveAddress_invoke.wsdl
    - LOCATION_invoke.wsdl
    - LOCATION_SYNC.VERSION_1_receive.wsdl
    - queryWithView_invoke.wsdl
    - SalesOrder.wsdl
    - SampleAccount.wsdl

8. Select **LOCATION_SYNC.VERSION_1_receive.wsdl** and click **OK**.

   The Create Partner Link dialog box is displayed.



Name: PSFT_PL1

WSDL File: ons/LOCATION_SYNC.VERSION_1_receive.wsdl?wsdl

Partner Link Type: LOCATION_SYNC.VERSION_1PartnerLinkType

My Role: LOCATION_SYNC.VERSION_1Role

Partner Role: ----- Not Specified -----

The **WSDL File** field displays the name and location of the selected WSDL file. The **Partner Link Type** field specifies the PartnerLink defined in the WSDL file.

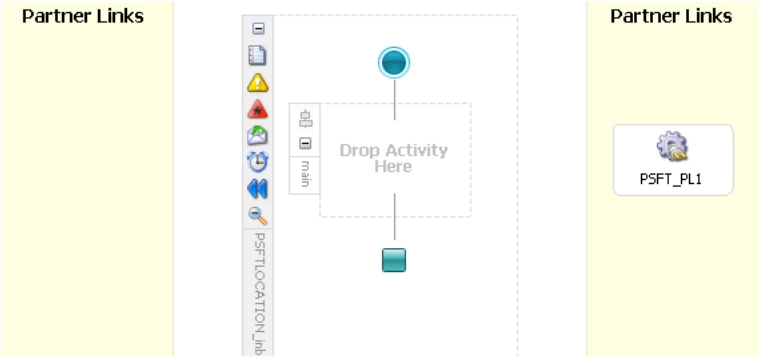
Perform the following steps:

a. From the **My Role** list, select the default value **LOCATION_SYNC.VERSION_1Role**.

b. Leave the **Partner Role** field unspecified.

9. Click **Apply**, and then **OK**.

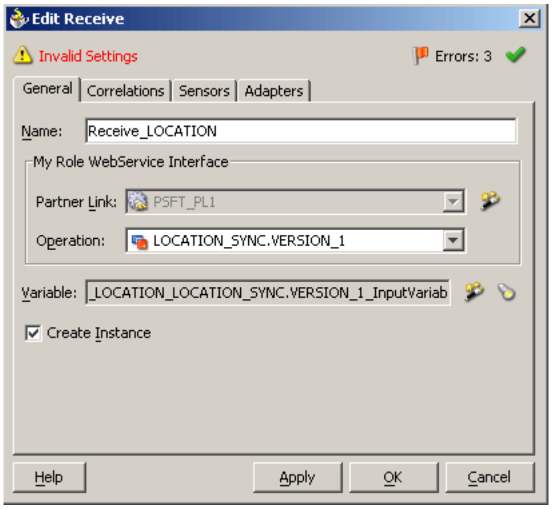
   The new PSFT_PL1 PartnerLink appears in the visual editor.



10. From the **Process Activities** pane on the right, drag a **Receive** activity to the visual editor and place it in the designated placeholder labeled **Drop Activity Here**.

11. Connect the Receive activity to the PSFT_PL1 PartnerLink.

    The Edit Receive dialog box is displayed.



Name: Receive_LOCATION

Partner Link: PSFT_PL1

Operation: LOCATION_SYNC.VERSION_1

Variable: _LOCATION_LOCATION_SYNC.VERSION_1_InputVariab

☑ Create Instance

    Perform the following steps:

    a. Specify a name for the Receive Activity, for example, **Receive_LOCATION**.

    b. Click the first icon to the right of the **Variable** field, then click **OK** in the Create Variable dialog box that is displayed.

    c. Verify that the **Create Instance** check box is selected.

12. Click **Apply**.

    The Receive dialog box should no longer display any warnings or errors.

13. Click **OK**.

14. Select **Save** from the **File** menu.

Deploying the BPEL Process for the LOCATION_SYNC.VERSION_1 Inbound Service

1. Right-click your process flow in the Applications - Navigator pane.

2. Select **Deploy**, then ***Your BPEL PM Server connection***, and then **Deploy to default domain**.

3. When prompted, enter your BPEL Process Manager password and click **OK**.

The deployment process starts automatically after you enter the correct password.

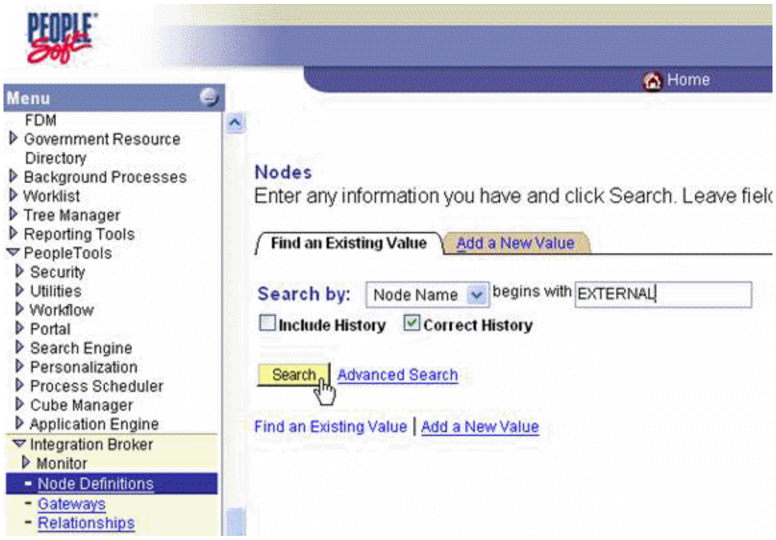
Runtime Configuration

The following topics describe how to trigger an event in PeopleSoft and verify event integration using OracleAS Adapter for PeopleSoft.

Triggering an Event in PeopleSoft to Test Event Runtime Integration

To trigger an event in PeopleSoft:

1. Log in to PeopleSoft, select **PeopleTools**, **Integration Broker**, and then **Node Definitions**.



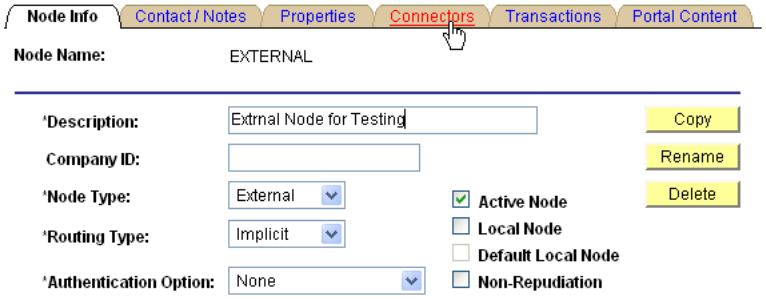
Nodes

Enter any information you have and click Search. Leave fiel

Search by: Node Name begins with EXTERNAL

□ Include History ☑ Correct History

2. Enter the node name you will be using, for example, `EXTERNAL`; then click **Search**.



Node Name: EXTERNAL

*Description: Extrnal Node for Testing

Company ID:

*Node Type: External

*Routing Type: Implicit

*Authentication Option: None

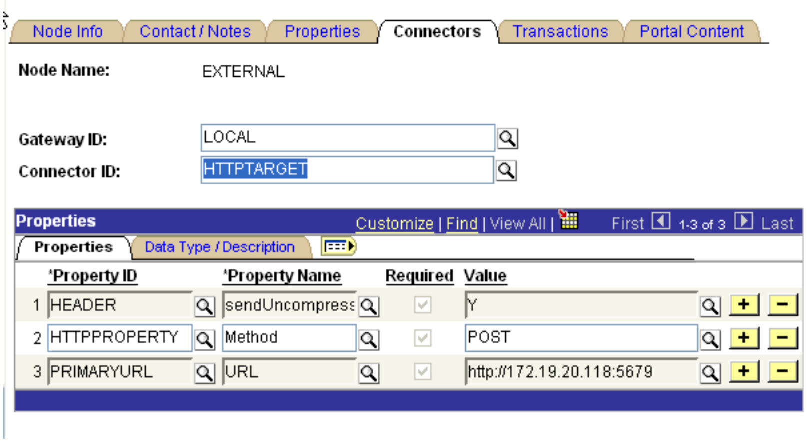
☑ Active Node

□ Local Node

□ Default Local Node

□ Non-Repudiation

3. Click the **Connectors** tab.



Node Name: EXTERNAL

Gateway ID: LOCAL

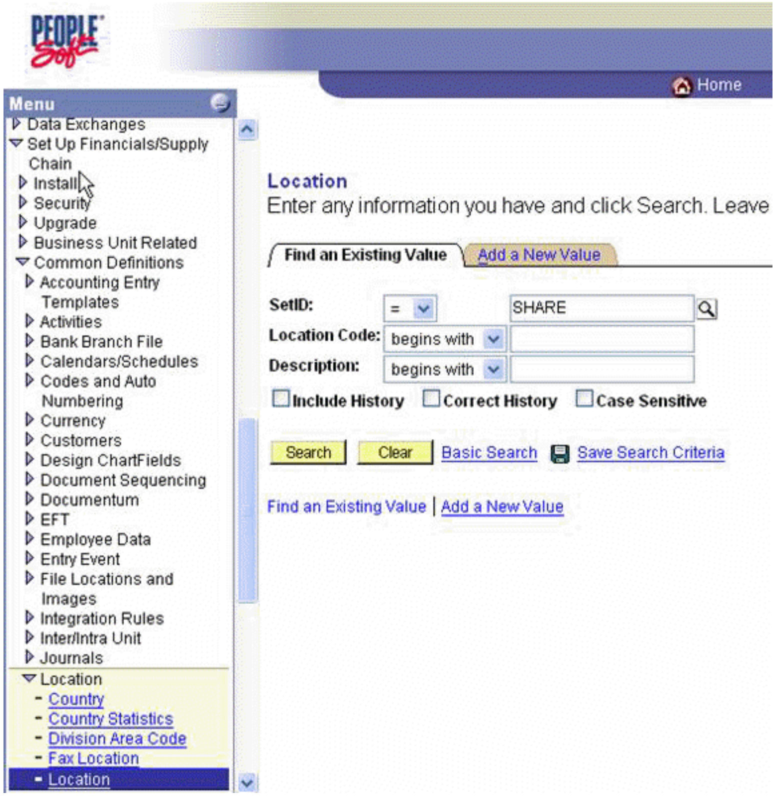
Connector ID: HTTPTARGET

Properties

	*Property ID	*Property Name	Required	Value
1	HEADER	sendUncompress	✓	Y
2	HTTPPROPERTY	Method	✓	POST
3	PRIMARYURL	URL	✓	http://172.19.20.118:5679

4. Enter **HTTPTARGET** in the **Connector ID** field; then enter the **listener URL** and its port in the **PRIMARYURL** field.

5. To save the configuration, click **Save**.



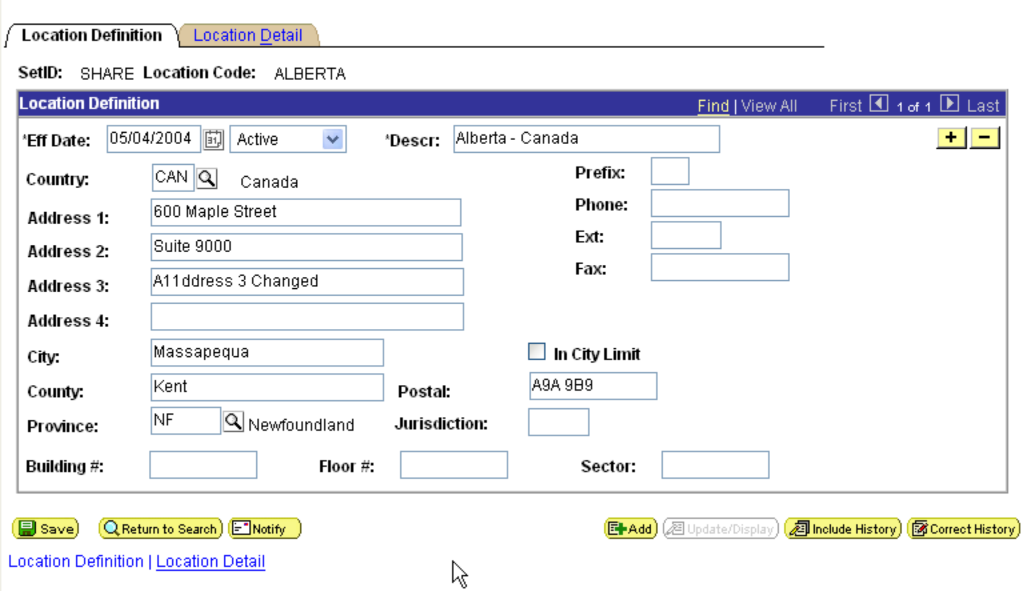
Location

Enter any information you have and click Search. Leave

SetID:	=	SHARE
Location Code:	begins with	
Description:	begins with	

□ Include History □ Correct History □ Case Sensitive

6. From **Menu** in the left pane, select **Set Up Financials/Supply Chain**, **Common Definitions**, **Location**, and then **Location**.



SetID: SHARE Location Code: ALBERTA

Location Definition Find | View All First 1 of 1 Last

*Eff Date: 05/04/2004 Active *Descr: Alberta - Canada

Country: CAN Canada

Address 1: 600 Maple Street

Address 2: Suite 9000

Address 3: A11ddress 3 Changed

Address 4:

City: Massapequa

County: Kent

Province: NF Newfoundland

Building #:

Prefix:

Phone:

Ext:

Fax:

□ In City Limit

Postal: A9A 9B9

Jurisdiction:

Floor #:

Sector:

7. To find the location record you want to update, click **Search**; then make the changes and click **Save**.

   When the change is saved, it triggers an event and sends the event to the listener.

Verifying the Results

To verify your results:

1. Log in to Oracle BPEL Console at

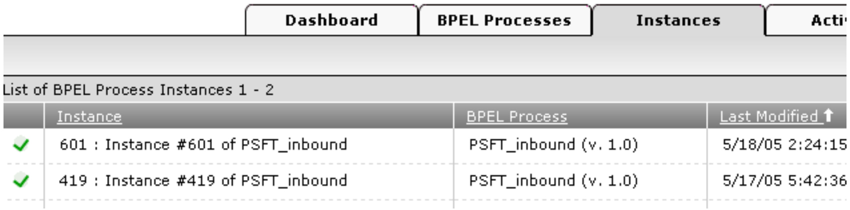
   `http://`*`host`*`:`*`port`*`/BPELConsole`

2. Enter the password for your BPEL domain.

   The default password is `bpel`.

3. Click the **Instances** tab.

   Recently received runtime events are displayed in the Instances tab.



List of BPEL Process Instances 1 - 2

	Instance	BPEL Process	Last Modified ⬆
✓	601 : Instance #601 of PSFT_inbound	PSFT_inbound (v. 1.0)	5/18/05 2:24:15
✓	419 : Instance #419 of PSFT_inbound	PSFT_inbound (v. 1.0)	5/17/05 5:42:36

4. Click an instance, then click **Audit** to see the received event message.

6

InterConnect Integration Examples

This chapter contains the following examples:

- Adapter Configuration in Application Explorer
- PeopleSoft Service Integration
- PeopleSoft Event Integration

The scenarios shown in this chapter require the following prerequisites.

Prerequisites

The following are installation and configuration requirements:

- OracleAS Adapter for PeopleSoft must be installed on Oracle Application Server.
- OracleAS Database adapter must be deployed and properly configured.
- OracleAS Integration InterConnect Adapter Plugin for EIS must be installed and running.

> **See Also:** *Oracle Application Server Adapter Installation Guide*

The examples in this chapter present the configuration steps necessary for demonstrating service and event integration with PeopleSoft. Prior to using this material, you must be familiar with the following:

- How to configure OracleAS Adapter for PeopleSoft for services and events. For more information, see Chapter 2, "Configuring OracleAS Adapter for PeopleSoft".
- How to configure OracleAS Integration InterConnect iStudio (iStudio) for service and event interactions. All required steps are shown in the examples.

> **See Also:** *Oracle Application Server Integration InterConnect User's Guide*

Overview of InterConnect Integration

InterConnect provides a comprehensive application integration framework. OracleAS Adapter for PeopleSoft used in conjunction with InterConnect enables you to seamlessly integrate enterprise software, eliminating the need to write custom code. Functional modeling, as opposed to custom coding solutions, allows for software reuse and reduces the complexity and management challenges that arise over the software lifecycle. This integration model consists of two components--high-level integration logic and low-level platform services.

Adapter integration with OracleAS InterConnect is a two-step process:

1. **Design Time:** OracleAS Adapter for PeopleSoft is configured in Application Explorer for services and events, as described in Chapter 2, "Configuring OracleAS Adapter for PeopleSoft". Integration logic is modeled in iStudio. Metadata are stored in repositories.
2. **Runtime:** The underlying platform treats this metadata as runtime instructions to enable the communication between participating applications.

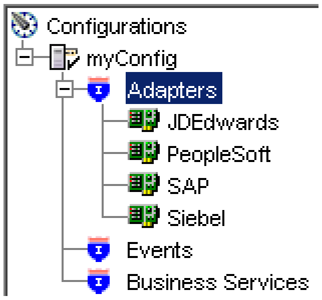
Adapter Configuration in Application Explorer

The following example describes how to establish a connection to your PeopleSoft system, and how to create a port and a channel for LOCATION_SYNC.VERSION_1.

Connecting to PeopleSoft and Viewing Application System Objects

1. In the left pane of Application Explorer, expand the **Adapters** node.

   The applications systems supported by Application Explorer appear as nodes based on the adapters that are installed.



2. Right-click the **PeopleSoft** node and select **Add Target**.

   The Add Target dialog box opens. Provide the following information:

   a. In the **Name** field, type a descriptive name, for example, PsoftTarget.

   b. In the **Description** field, type a description for the target (optional).

   c. From the **Target Type** list, select a target type.

      The default value is Application Server.

3. Click **OK**.

   The Application Server dialog box opens where you must specify connection information for PeopleSoft and the application server that is hosting PeopleSoft.

4. Verify your connection parameters. Provide the required password.
5. Right-click the target name and select **Connect**.

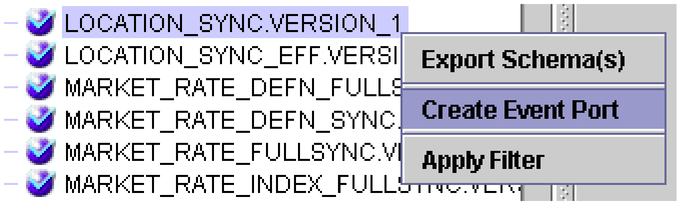
   The x icon disappears, indicating that the node is connected.

6. From the expanded Adapter list, expand the **PeopleSoft** node, expand **Messages**, and then select **LOCATION_SYNC.VERSION_1**.

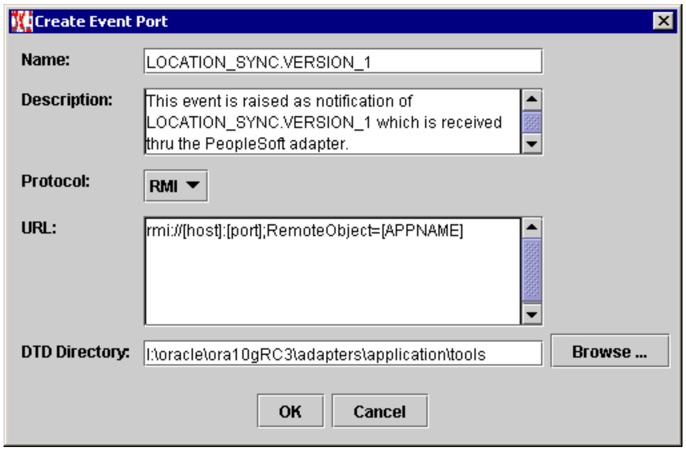
Creating an Event Port for LOCATION.SYNC.VERSION1

To create an event port for LOCATION_SYNC.VERSION_1:

1. Right-click the **LOCATION_SYNC.VERSION_1** node and select **Create Event Port**.



The Create Event Port dialog box is displayed.



Name: LOCATION_SYNC.VERSION_1

Description: This event is raised as notification of LOCATION_SYNC.VERSION_1 which is received thru the PeopleSoft adapter.

Protocol: RMI

URL: rmi://[host]:[port];RemoteObject=[APPNAME]

DTD Directory: I:\oracle\ora10gRC3\adapters\application\tools

Perform the following steps:

a. Type a name for the event port and provide a brief description.

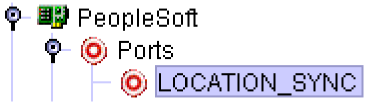
b. From the list, select the required disposition, for example, RMI.

c. Type the disposition URL.

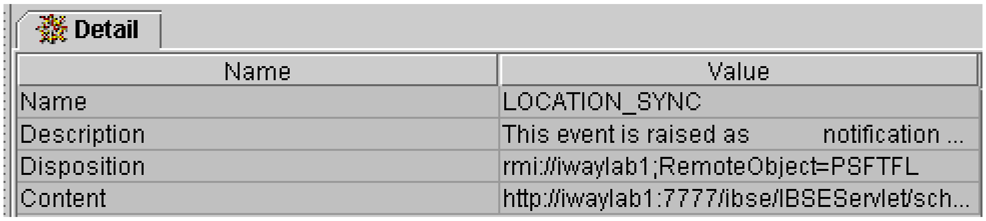
d. Type (or browse to) the path containing the DTD directory.

2. Click **OK**.

The port appears under the ports node in the left pane.



In the right pane, a table appears that summarizes the information associated with the event port you created.



Name	Value
Name	LOCATION_SYNC
Description	This event is raised as notification ...
Disposition	rmi://iwaylab1;RemoteObject=PSFTFL
Content	http://iwaylab1:7777/ibse/IBSEServlet/sch...

You can now associate the event port with a channel.

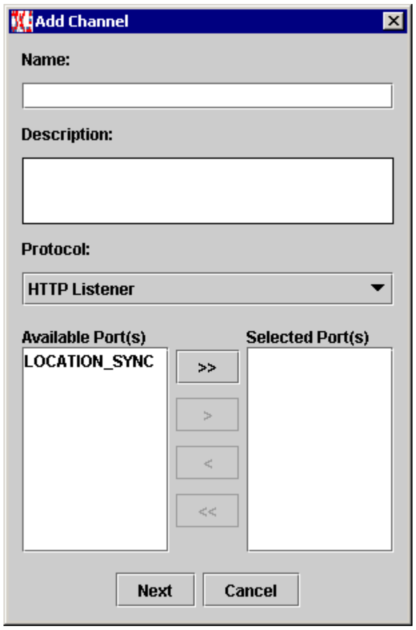
Creating a Channel

To create a channel:

1. In the left pane, click the **Events** node.
2. Expand the **PeopleSoft** node.

   The ports and channels nodes appear in the left pane.
3. Right-click **Channels** and select **Add Channel**.

   The Add Channel dialog box opens.



   Perform the following steps:

   a. Type a name for the channel, for example, **TEST_CHANNEL**.

   b. Type a brief description.

   c. From the **Protocol** list, select **HTTP Listener**.

   d. Select an event port from the list of available ports. To select more than one, hold down the **Ctrl** key and click the ports.

   e. To transfer the ports to the list of selected ports, click the **double right (>>)** arrow button.
4. Click **Next**.
5. When the Basic dialog box opens, type the system information as specified in the following table:

Parameter	Description
Listener port	Port on which to listen for PeopleSoft event data.
Https	For a secure HTTP connection, select the **Https** check box.

Parameter	Description
Synchronization Type	Choose from the following synchronization options: - REQUEST_RESPONSE - REQUEST_ACK **Important:** The PeopleSoft channel will not work if the synchronization type is set to REQUEST.

6. Click **OK**.

   The channel appears under the channels node in the left pane.

   An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

7. Right-click the channel node and select **Start**.

   The channel you created becomes active. The X over the icon in the left pane disappears.

   To stop the channel, right-click the connected channel node and select **Stop**.

PeopleSoft Service Integration

This topic illustrates PeopleSoft service integration. Design-time and runtime configuration procedures are described.

Design-Time Configuration

The following procedures describe how to start the InterConnect repository, create a common view, and define invoked and implemented procedures. Then, it describes how to export PL/SQL code from iStudio.

Starting the Repository

To start the InterConnect repository, double-click the `start.bat` file located in the following directory:

```
InterConnect_home\repository\start.bat
```

Where `InterConnect_home` is the directory where InterConnect is installed, for example, `C:\OraHome_1\integration\interconnect\`.

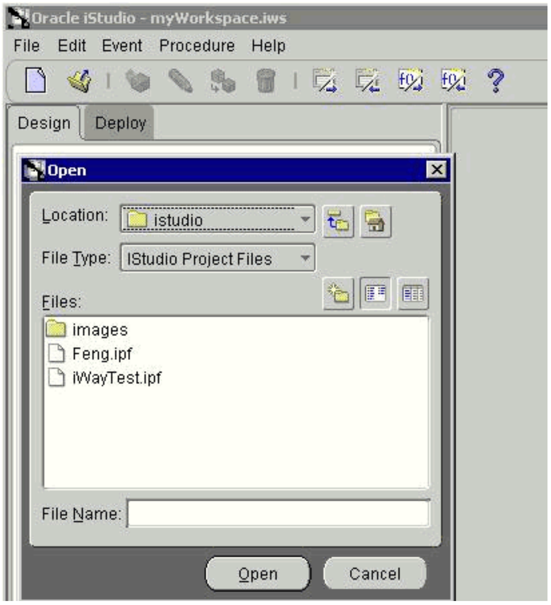
Creating a Common View

To create a Common View:

1. Start iStudio by double-clicking the `start.bat` file located in the following directory:

   ```
   InterConnect_home\istudio\iStudio.bat
   ```

iStudio starts.


Oracle iStudio - myWorkspace.iws
File Edit Event Procedure Help
Design
Deploy
Open
Files:
Open
Cancel

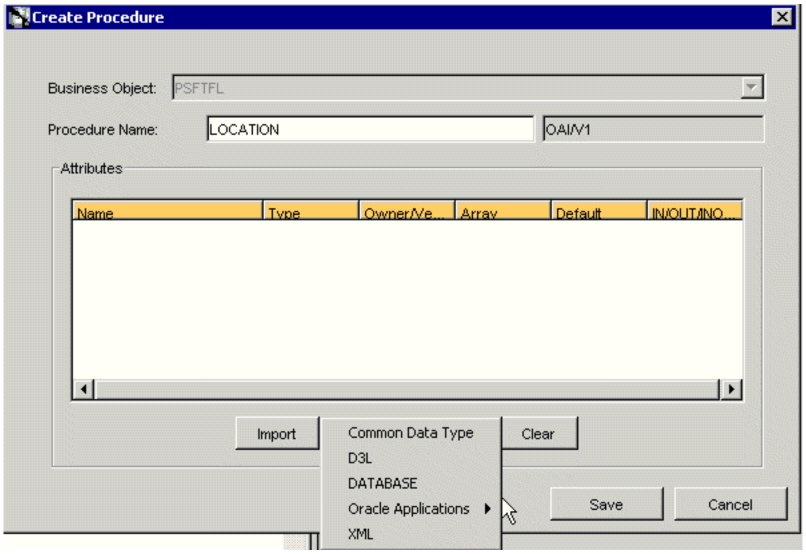
Location: istudio

File Type: IStudio Project Files

images
Feng.ipf
iWayTest.ipf

File Name:

2. Open a project.
3. Open **Common Views** and **Business Objects**.
4. Create a Business Object called **PSFTFL**.



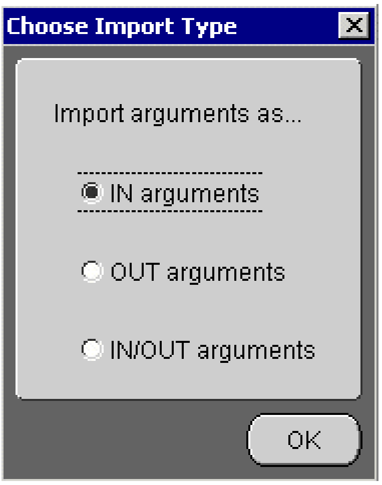
Business Object: PSFTFL

Procedure Name: LOCATION OAI/V1

Name	Type	Owner/Ve...	Array	Default	IN/OUT/INO...

5. Create a new procedure under `PSFTFL` and type `LOCATION` as the procedure name.
6. Open the DTD generated from Application Explorer and load it.

   The Choose Root Element dialog box opens.
7. Select the root element, `PS8`, for this example.
8. Click **OK**.



Choose Import Type

Import arguments as...

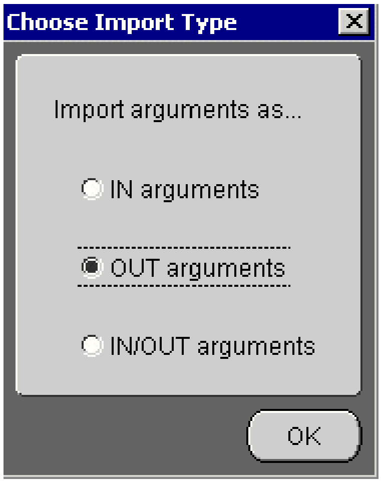
● IN arguments

○ OUT arguments

○ IN/OUT arguments

9. Select **IN arguments** as the import type for the request DTD and click **OK**.
10. Import the response DTD, select the root element, and click **OK**.

    The Choose Import Type dialog box opens.



Choose Import Type

Import arguments as...

○ IN arguments

● OUT arguments

○ IN/OUT arguments

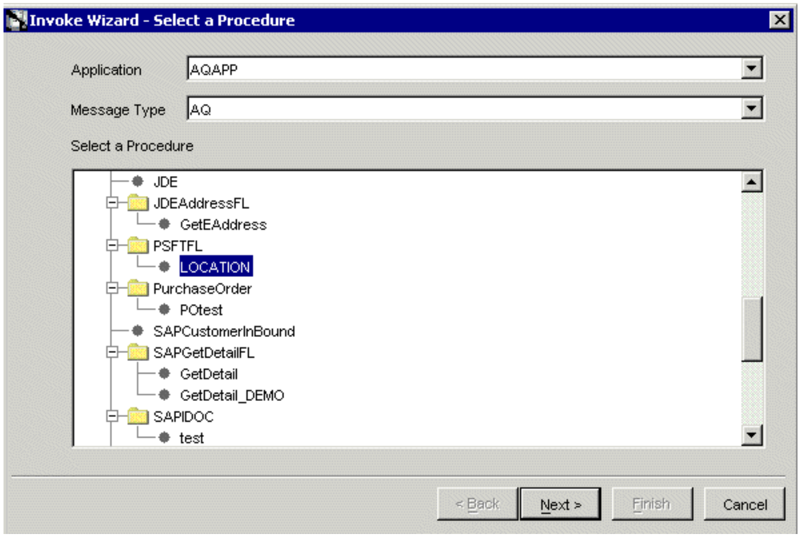
11. Select **OUT arguments** as the import type for the response DTD and click **OK**.
12. To save the new procedure, click **Save**.

Creating an Invoked Procedure

To create an invoked procedure:

1. Create a new application called **AQAPP**.
2. Right-click **Invoked Procedures** and select **New**.

   The Invoke Wizard - Select a Procedure window opens.



Invoke Wizard - Select a Procedure

Application: AQAPP

Message Type: AQ

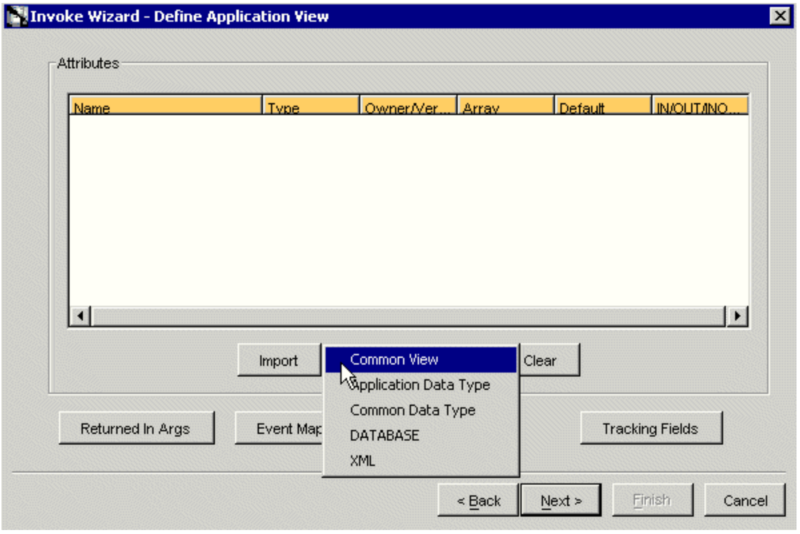
Perform the following steps:

a. From the **Message Type** list, select **AQ**.

b. Expand the `PSFTFL` business object as the event and select **LOCATION**.

3. Click **Next**.

The Invoke Wizard - Define Application View window opens.



Invoke Wizard - Define Application View

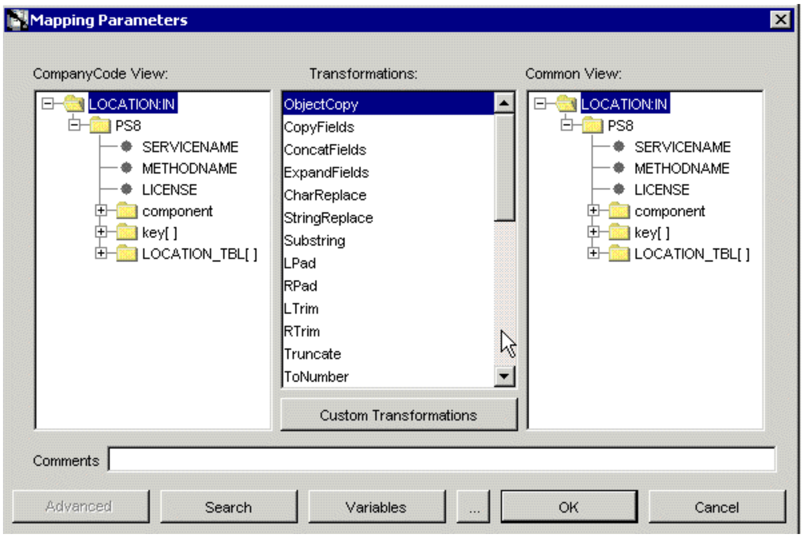
Perform the following steps:

a. Click **Import**.

b. Select **Common View**.

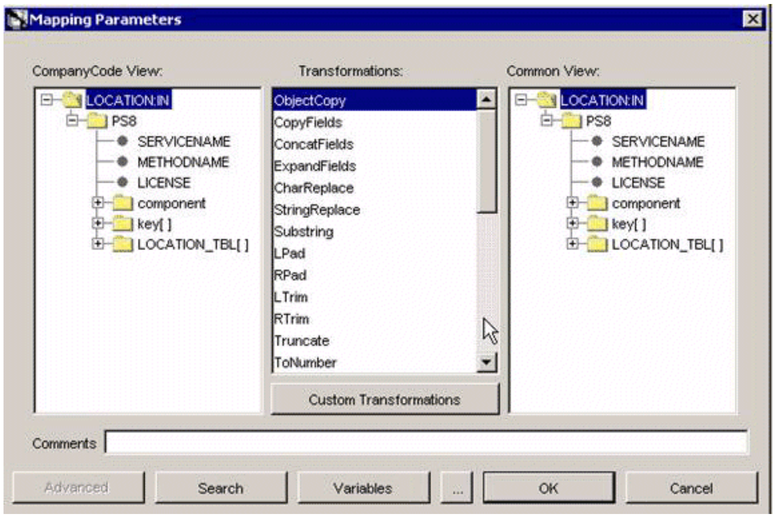
4. Click **Next**.

5. Click **New** to create a mapping between the Common View and the Application View for the IN parameters.



In this example, the Application View and the Common View have the same structure. All the attributes can be mapped by using ObjectCopy Transformation.

6. Click **Apply**, and then **OK**.

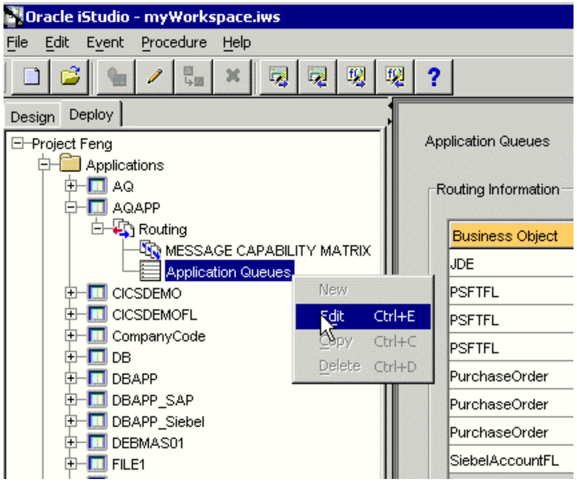


7. Click **Finish**.

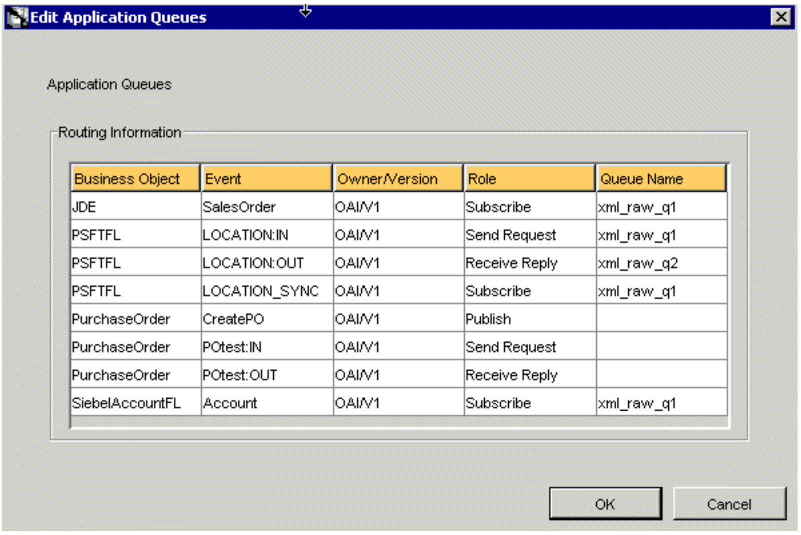
   You have completed defining the application definition for the invoked procedure.

Defining Queues for AQ

You must create and specify a queue for Advanced Queuing (AQ):



1. To specify the queue in iStudio, select the **Deploy** tab, **Project**, **Application**, **AQAPP**, **Routing**, and **Application Queues**.
2. Right-click **Application Queues** and select **Edit**.



Business Object	Event	Owner/Version	Role	Queue Name
JDE	SalesOrder	OAI/V1	Subscribe	xml_raw_q1
PSFTFL	LOCATION:IN	OAI/V1	Send Request	xml_raw_q1
PSFTFL	LOCATION:OUT	OAI/V1	Receive Reply	xml_raw_q2
PSFTFL	LOCATION_SYNC	OAI/V1	Subscribe	xml_raw_q1
PurchaseOrder	CreatePO	OAI/V1	Publish	
PurchaseOrder	POtest:IN	OAI/V1	Send Request	
PurchaseOrder	POtest:OUT	OAI/V1	Receive Reply	
SiebelAccountFL	Account	OAI/V1	Subscribe	xml_raw_q1

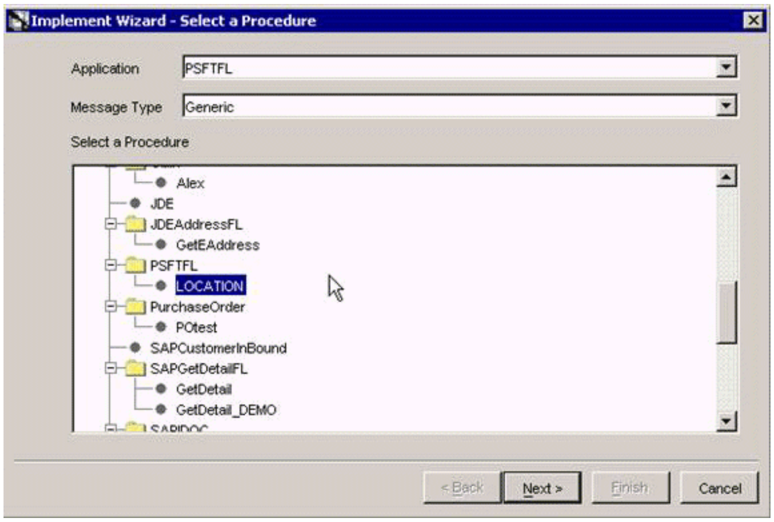
3. In the **Queue Name** column, type the queue name, in this example, **xml_raw_ql** for `PSFTFL LOCATION:IN` event, and **xml_raw_q2** for `PSFTFL LOCATION:OUT` event.
4. To finish, click **OK**.
5. To create xml_raw_ql and xml_raw_q2 queues, execute the following SQL under the appropriate schema, in this example, the AQAPP schema.

```
EXECUTE dbms_aqadm.create_queue_table (queue_table => 'RawMsgs_qtab', queue_
payload_type => 'RAW', multiple_consumers => FALSE);
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q1', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q1');
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q2', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q2');
```

Implementing the Procedure

To implement the procedure:

1. Create a new application called `PSFTFL`, expand the application, right-click **Implemented Procedure**, and select **New**.



Implement Wizard - Select a Procedure

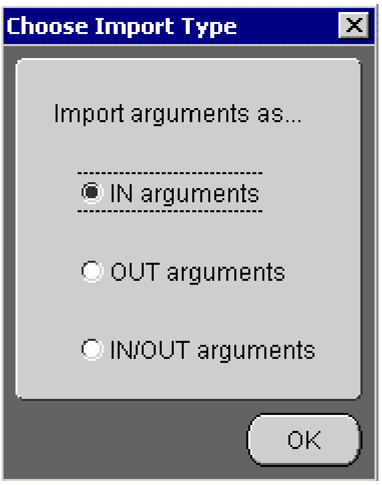
Application: PSFTFL

Message Type: Generic

Select a Procedure

2. From the **Message Type** list, select **Generic**.
3. Select **LOCATION** under **PSFTFL** as the procedure.
4. Click **Next**.
5. Open the DTD generated from Application Explorer and load it.

   The Choose Root Element dialog box is displayed.
6. Select the root element, `PS8`, for this example.
7. Click **OK**.



Choose Import Type

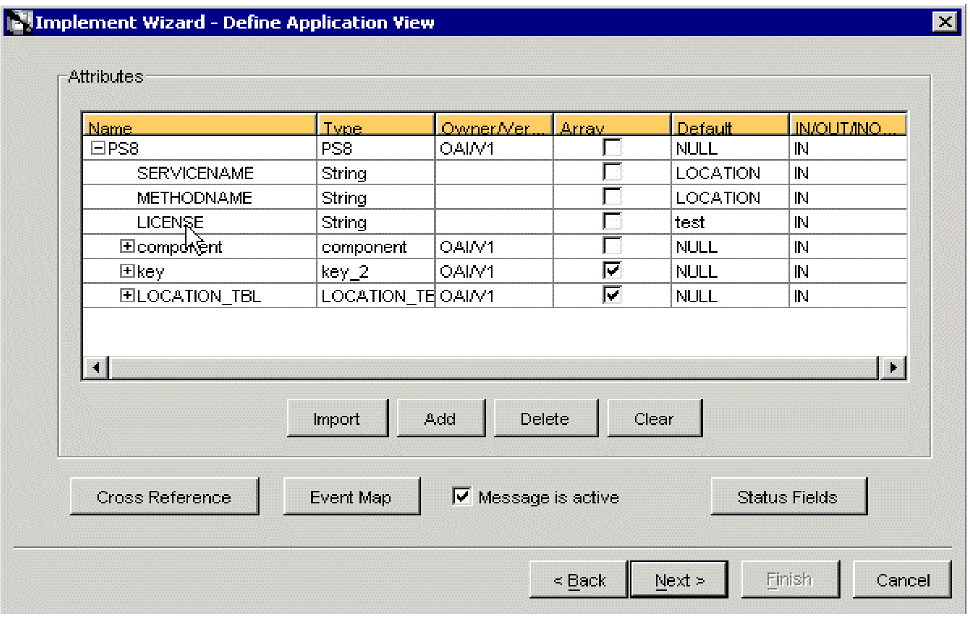
Import arguments as...

● IN arguments

○ OUT arguments

○ IN/OUT arguments

8. Select **IN arguments** as the import type for the request DTD and click **OK**.
9. In the **Name** field, type the root element of the request DTD if it is not populated automatically after importing the request DTD.



Implement Wizard - Define Application View

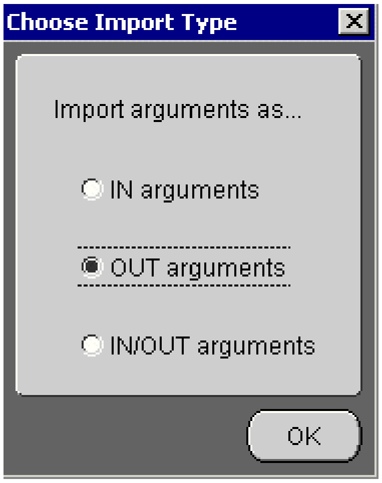
Attributes

Name	Type	Owner/Ver...	Array	Default	IN/OUT/INO...
⊟PS8	PS8	OAI/V1	☐	NULL	IN
SERVICENAME	String		☐	LOCATION	IN
METHODNAME	String		☐	LOCATION	IN
LICENSE	String		☐	test	IN
⊞component	component	OAI/V1	☐	NULL	IN
⊞key	key_2	OAI/V1	☑	NULL	IN
⊞LOCATION_TBL	LOCATION_TE	OAI/V1	☑	NULL	IN

☑ Message is active

10. Import the response DTD, select `PS8` as the root element, and then click **OK**.

11. Select OUT arguments as the Import Type for the response.



Choose Import Type

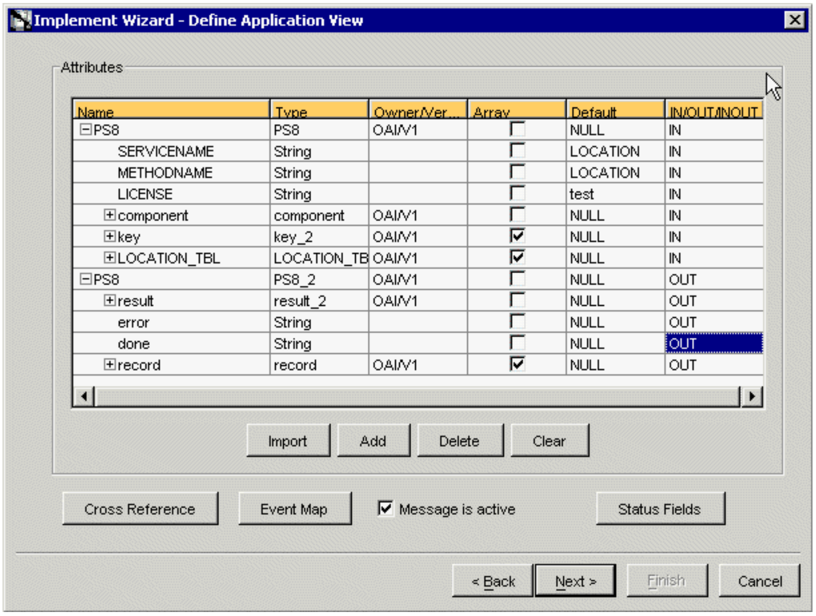
Import arguments as...

○ IN arguments

● OUT arguments

○ IN/OUT arguments

You have now imported both request and response DTDs into iStudio.



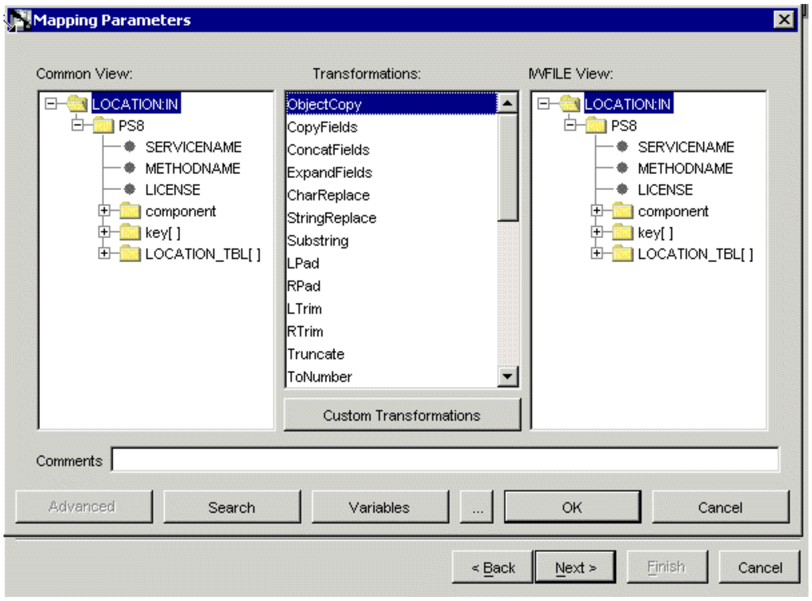
Implement Wizard - Define Application View

Attributes

Name	Type	Owner/Ver...	Array	Default	IN/OUT/INOUT
⊟PS8	PS8	OAI/V1	□	NULL	IN
SERVICENAME	String		□	LOCATION	IN
METHODNAME	String		□	LOCATION	IN
LICENSE	String		□	test	IN
⊞component	component	OAI/V1	□	NULL	IN
⊞key	key_2	OAI/V1	☑	NULL	IN
⊞LOCATION_TBL	LOCATION_TB	OAI/V1	☑	NULL	IN
⊟PS8	PS8_2	OAI/V1	□	NULL	OUT
⊞result	result_2	OAI/V1	□	NULL	OUT
error	String		□	NULL	OUT
done	String		□	NULL	OUT
⊞record	record	OAI/V1	☑	NULL	OUT

☑ Message is active

12. To define a mapping between the Application View and the Common View, click **Next** and then **New**.

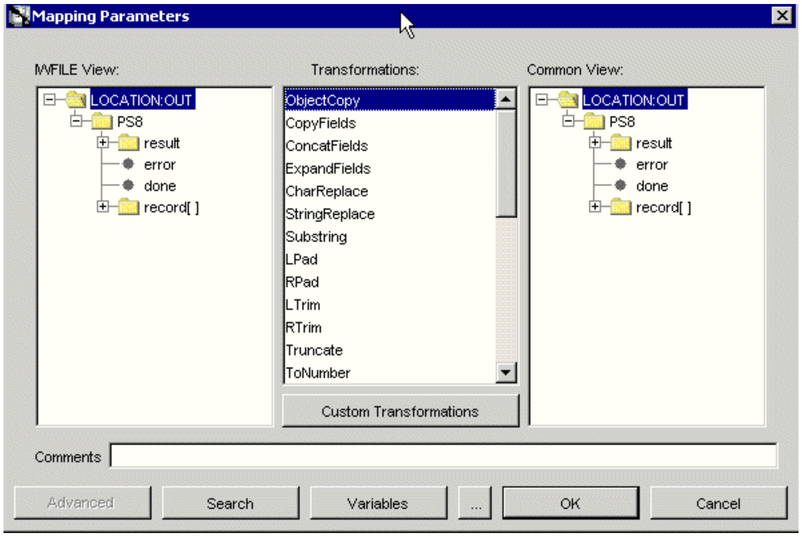


Mapping Parameters

Because the structures are the same, use ObjectCopy transformation for the mapping.

13. Click **Apply** and then **OK**.

14. Click **Next** to define the mapping for OUT arguments, using the ObjectCopy transformation.



15. To complete the definition of the implemented procedure, click **Next** and then **Finish**.

Editing the adapter.ini File

The `adapter.ini` file located under *`InterConnect_home`*`\adapters\`*`appname`* may require editing as described in the following procedure. If the following information is missing in `adapter.ini`, you must add it manually.

1. Open the `adapter.ini` file located under

   ```
   InterConnect_home\adapters\appname
   ```

   Where `appname` is the name of the application you created.

2. Add the following two lines to `adapter.ini`:

   ```
   // Bridge class
   bridge_class=com.iwaysoftware.iwbridge.IWBridge

   // IBSE URL
   ibse_url=http://hostname:port/ibse/IBSEServlet/XDSOAPRouter
   ```

   Where `hostname` is the name of the Oracle Application Server host and `port` is the server port number. The default server port number for Oracle Application Server is 7777.

Runtime Configuration

The following topic describes how to verify service integration using OracleAS Adapter for PeopleSoft.

Verifying Service Integration

To verify service integration:

1. Start Oracle Application Server or ensure that the server is running.
2. If necessary, restart OC4J by issuing the following command:

   ```
   OracleAS_home\opmn\bin\opmnctl  stopproc process-type=home
   OracleAS_home\opmn\bin\opmnctl  startproc process-type=home
   ```

3. Check the status of OC4J by issuing the following command:

```
OracleAS_home\opmn\bin\opmnctl  status
```

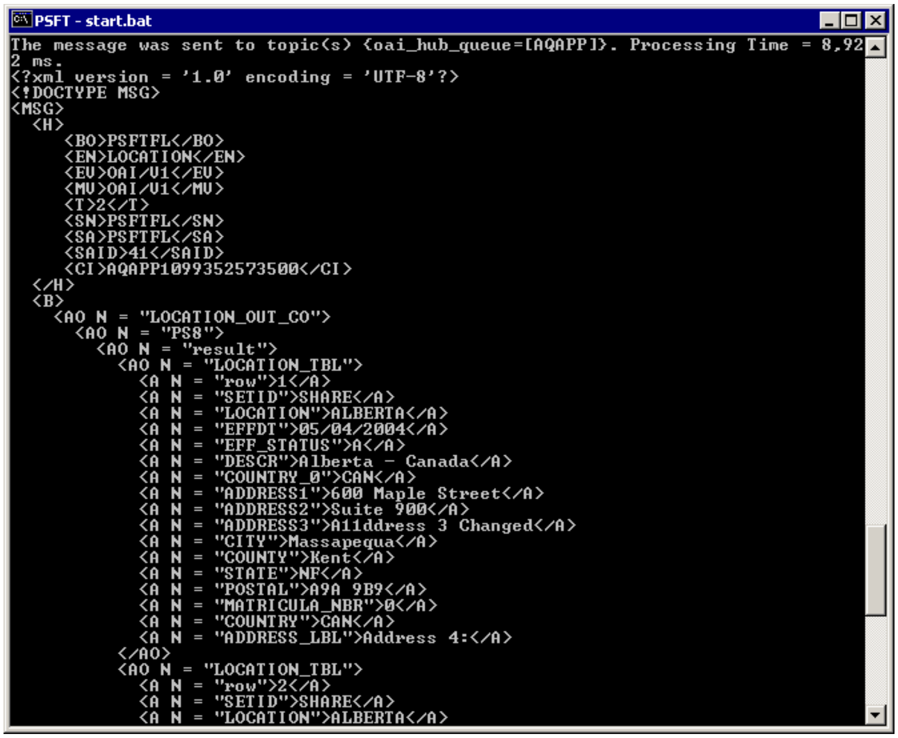
4. Start the adapter using `start.bat`:

```
InterConnect_home\adapters\PSFTFL\start.bat
InterConnect_home\adapters\AQAPP\start.bat
```

5. Log on to SQL*Plus with `AQAPP`, in this example, and issue the following command:

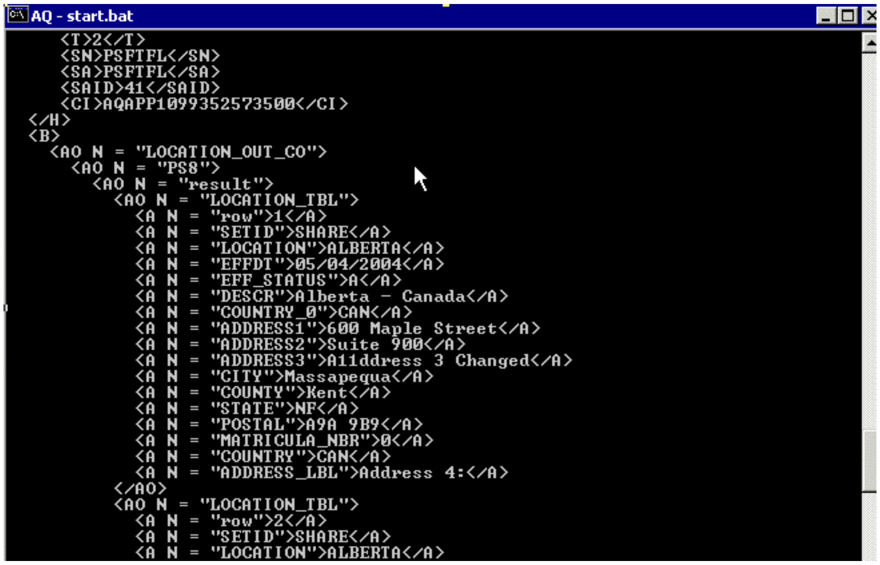
```
DECLARE
      enqueue_options     dbms_aq.enqueue_options_t;
      message_properties  dbms_aq.message_properties_t;
      msgid               RAW(16);
      payload             RAW(5000);
BEGIN
      payload := utl_raw.cast_to_raw('<?xml version="1.0" encoding="UTF-8"?>
<PS8 SERVICENAME="LOCATION" METHODNAME="LOCATION" LICENSE="test">
<component perform="browse">LOCATION</component>
<key name="SETID">SHARE</key>
<key name="LOCATION">ALBERTA</key>
</PS8>');
      dbms_aq.enqueue(queue_name         => 'xml_raw_q1',
                      enqueue_options    => enqueue_options,
                      message_properties => message_properties,
                      payload            => payload,
                      msgid              => msgid);
      COMMIT;
END;
/
```

The following image shows the `PSFTFL` example. It receives a reply from PeopleSoft and returns the reply to the hub.



```
The message was sent to topic(s) {oai_hub_queue=[AQAPP]}. Processing Time = 8,92
2 ms.
<?xml version = '1.0' encoding = 'UTF-8'?>
<!DOCTYPE MSG>
<MSG>
  <H>
     <BO>PSFTFL</BO>
     <EN>LOCATION</EN>
     <EV>OAI/V1</EV>
     <MV>OAI/V1</MV>
     <T>2</T>
     <SN>PSFTFL</SN>
     <SA>PSFTFL</SA>
     <SAID>41</SAID>
     <CI>AQAPP1099352573500</CI>
  </H>
  <B>
    <AO N = "LOCATION_OUT_CO">
      <AO N = "PS8">
        <AO N = "result">
          <AO N = "LOCATION_TBL">
            <A N = "row">1</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
            <A N = "EFFDT">05/04/2004</A>
            <A N = "EFF_STATUS">A</A>
            <A N = "DESCR">Alberta - Canada</A>
            <A N = "COUNTRY_0">CAN</A>
            <A N = "ADDRESS1">600 Maple Street</A>
            <A N = "ADDRESS2">Suite 900</A>
            <A N = "ADDRESS3">A11ddress 3 Changed</A>
            <A N = "CITY">Massapequa</A>
            <A N = "COUNTY">Kent</A>
            <A N = "STATE">NF</A>
            <A N = "POSTAL">A9A 9B9</A>
            <A N = "MATRICULA_NBR">0</A>
            <A N = "COUNTRY">CAN</A>
            <A N = "ADDRESS_LBL">Address 4:</A>
          </AO>
          <AO N = "LOCATION_TBL">
            <A N = "row">2</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
```

The following image shows the AQ example. It receives a reply from the hub and writes the data to the database table.



```
     <T>2</T>
     <SN>PSFTFL</SN>
     <SA>PSFTFL</SA>
     <SAID>41</SAID>
     <CI>AQAPP1099352573500</CI>
  </H>
  <B>
    <AO N = "LOCATION_OUT_CO">
      <AO N = "PS8">
        <AO N = "result">
          <AO N = "LOCATION_TBL">
            <A N = "row">1</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
            <A N = "EFFDT">05/04/2004</A>
            <A N = "EFF_STATUS">A</A>
            <A N = "DESCR">Alberta - Canada</A>
            <A N = "COUNTRY_0">CAN</A>
            <A N = "ADDRESS1">600 Maple Street</A>
            <A N = "ADDRESS2">Suite 900</A>
            <A N = "ADDRESS3">A11ddress 3 Changed</A>
            <A N = "CITY">Massapequa</A>
            <A N = "COUNTY">Kent</A>
            <A N = "STATE">NF</A>
            <A N = "POSTAL">A9A 9B9</A>
            <A N = "MATRICULA_NBR">0</A>
            <A N = "COUNTRY">CAN</A>
            <A N = "ADDRESS_LBL">Address 4:</A>
          </AO>
          <AO N = "LOCATION_TBL">
            <A N = "row">2</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
```

PeopleSoft Event Integration

This topic illustrates PeopleSoft event integration. Design-time and runtime configuration is described.

Design-Time Configuration

The following procedures describe how to start the InterConnect repository, create a common view, and then define invoked and implemented procedures. Then, it describes how to export PL/SQL code from iStudio.

Starting the Repository

To start the InterConnect repository, double-click the `start.bat` file located in the following directory:

```
InterConnect_home\repository\start.bat
```

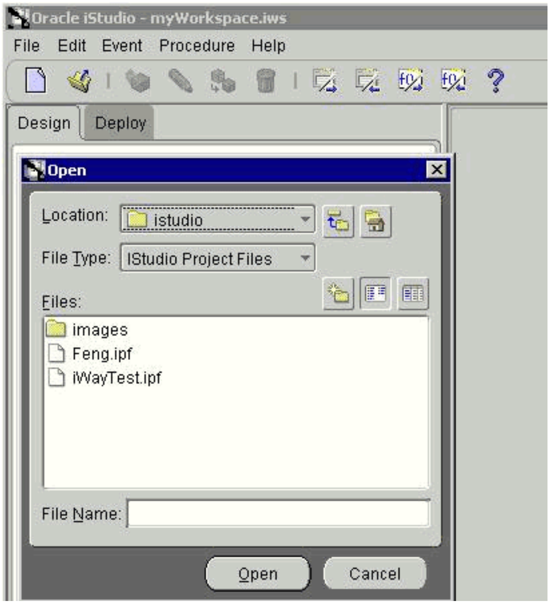
Creating a Common View

To create a Common View:

1. Start Oracle iStudio by double-clicking the `start.bat` file located in the following directory:

   ```
   InterConnect_home\istudio\iStudio.bat
   ```

   iStudio opens.



Oracle iStudio - myWorkspace.iws
File Edit Event Procedure Help
Design
Deploy
Open
Open
Cancel

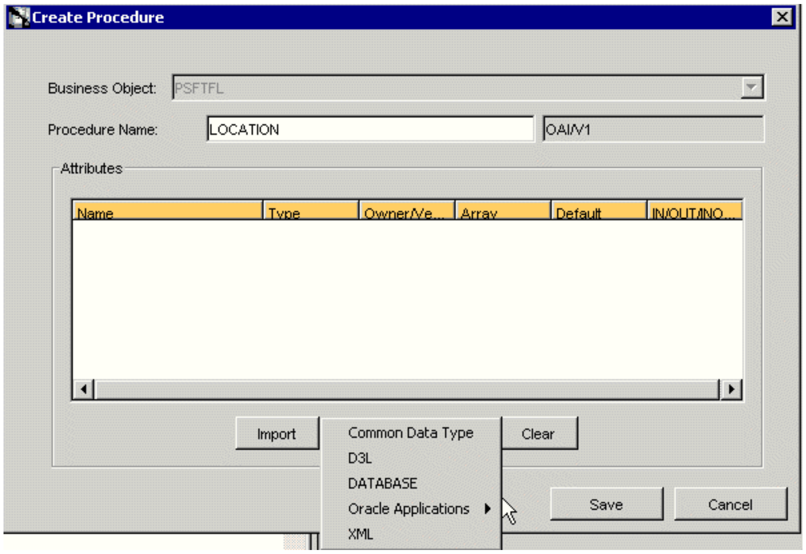
Location: istudio

File Type: IStudio Project Files

Files:
- images
- Feng.ipf
- iWayTest.ipf

File Name:

2. Open a project.
3. Open **Common Views** and **Business Objects**.
4. Create a Business Object called `PSFTFL`.

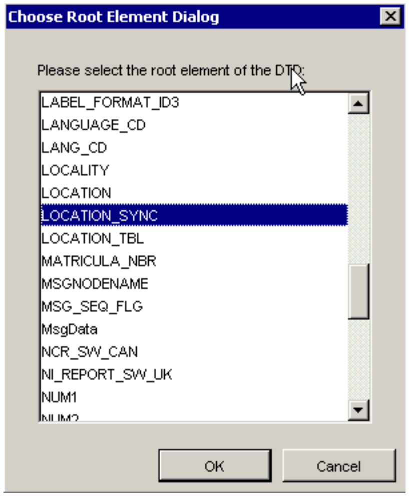


Business Object: PSFTFL

Procedure Name: LOCATION OAI/V1

5. Create a new procedure under **PSFTFL** and type `LOCATION` as the procedure name.
6. Open the DTD generated from Application Explorer and load it.

   The Choose Root Element dialog box opens.

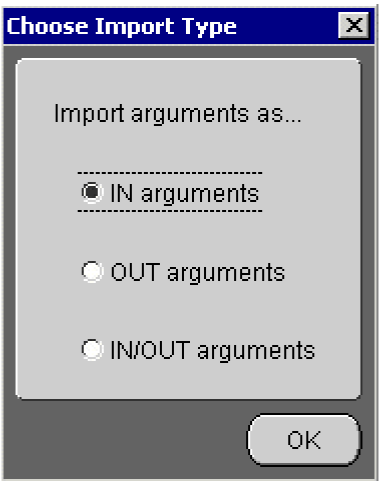


Choose Root Element Dialog

Please select the root element of the DTD:

LABEL_FORMAT_ID3
LANGUAGE_CD
LANG_CD
LOCALITY
LOCATION
LOCATION_SYNC
LOCATION_TBL
MATRICULA_NBR
MSGNODENAME
MSG_SEQ_FLG
MsgData
NCR_SW_CAN
NI_REPORT_SW_UK
NUM1
NUM2

7. Select the root element, `PS8`, for this example.
8. Click **OK**.



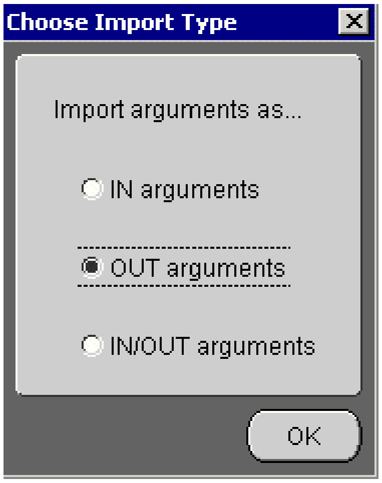
Choose Import Type

Import arguments as...

● IN arguments
○ OUT arguments
○ IN/OUT arguments

9. Select **IN arguments** as the import type for the request DTD and click **OK**.
10. Import the response DTD, select the root element, and click **OK**.

    The Choose Import Type dialog box opens.



Choose Import Type

Import arguments as...

○ IN arguments
● OUT arguments
○ IN/OUT arguments

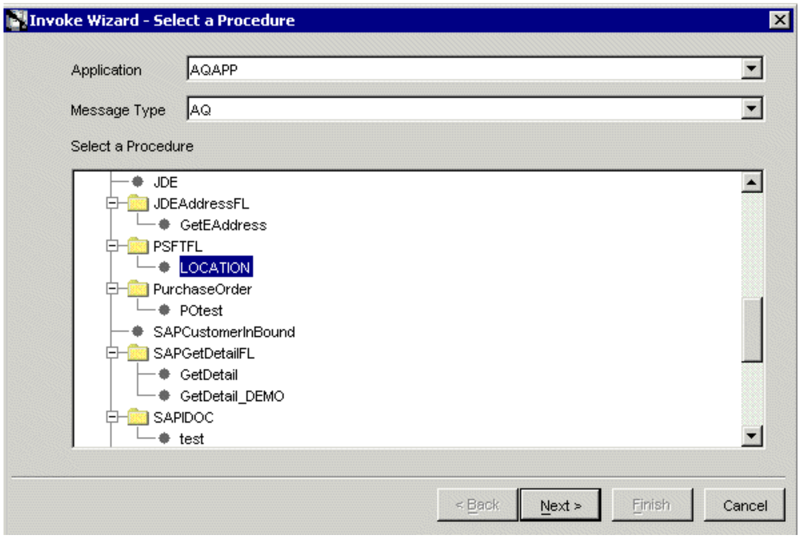
11. Select **OUT arguments** as the import type for the response DTD and click **OK**.
12. To save the new procedure, click **Save**.

Creating an Invoked Procedure

To create an invoked procedure:

1. Create a new application called **AQAPP**.
2. Right-click **Invoked Procedures** and select **New**.

   The Invoke Wizard - Select a Procedure window opens.



Invoke Wizard - Select a Procedure

Application: AQAPP

Message Type: AQ

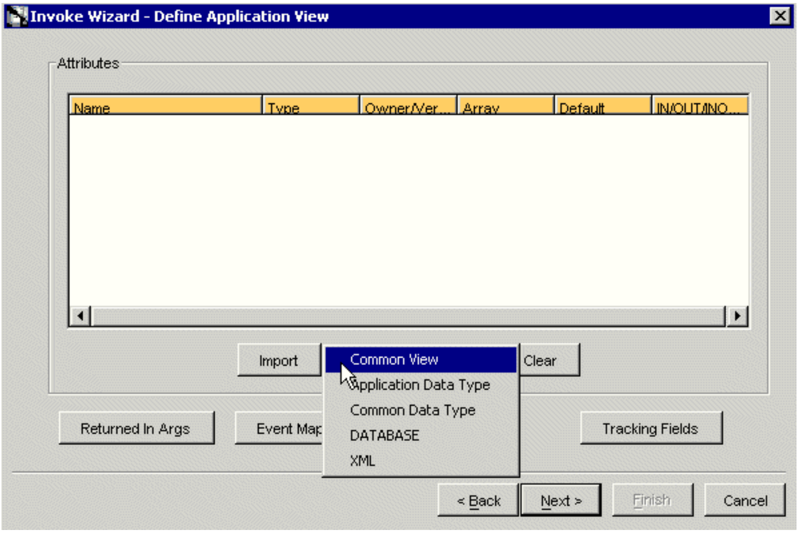
   Perform the following steps:

   a. From the **Message Type** list, select **AQ**.

   b. Expand the `PSFTFL` business object as the event and select **LOCATION**.

3. Click **Next**.

   The Invoke Wizard - Define Application View window opens.

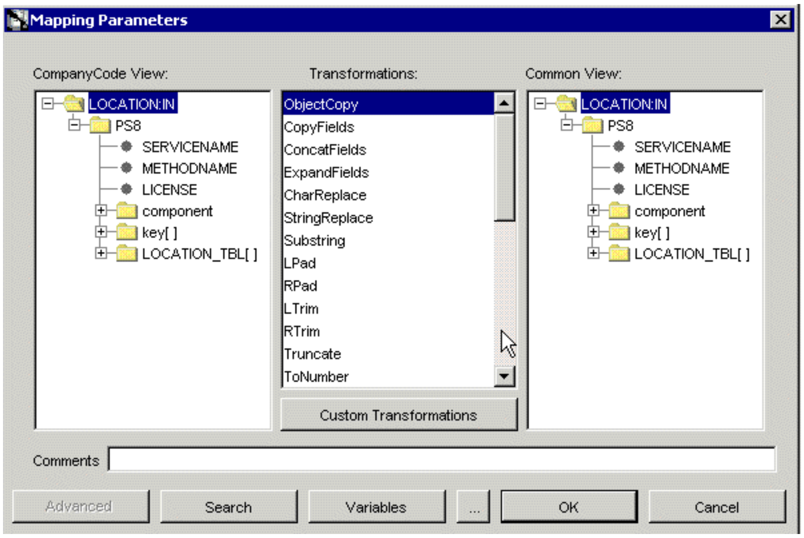


Invoke Wizard - Define Application View

   Perform the following steps:

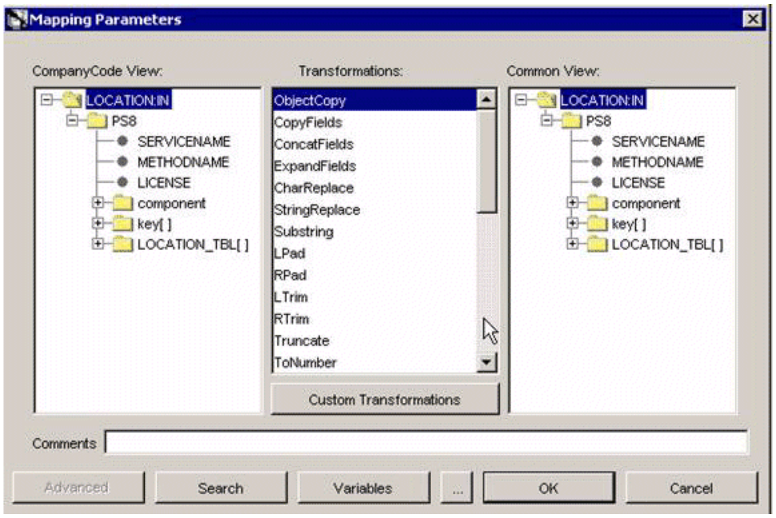
  a. Click **Import**.
  b. Select **Common View**.

4. Click **Next**.
5. Click **New** to create a mapping between the Common View and the Application View for the IN parameters.



In this example, the Application View and the Common View have the same structure. All the attributes can be mapped by using ObjectCopy Transformation.

6. Click **Apply** and then **OK**.

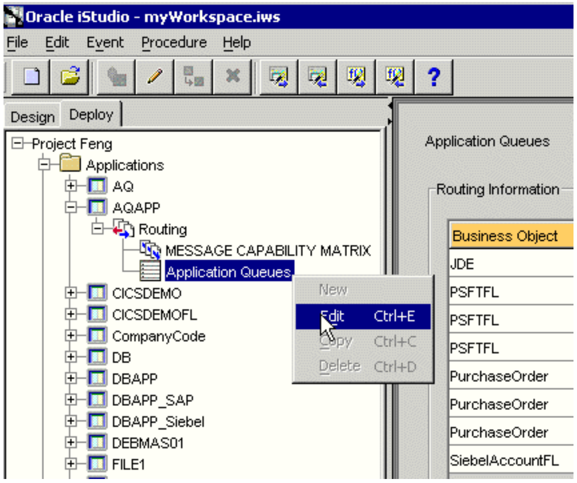


7. Click **Finish**.

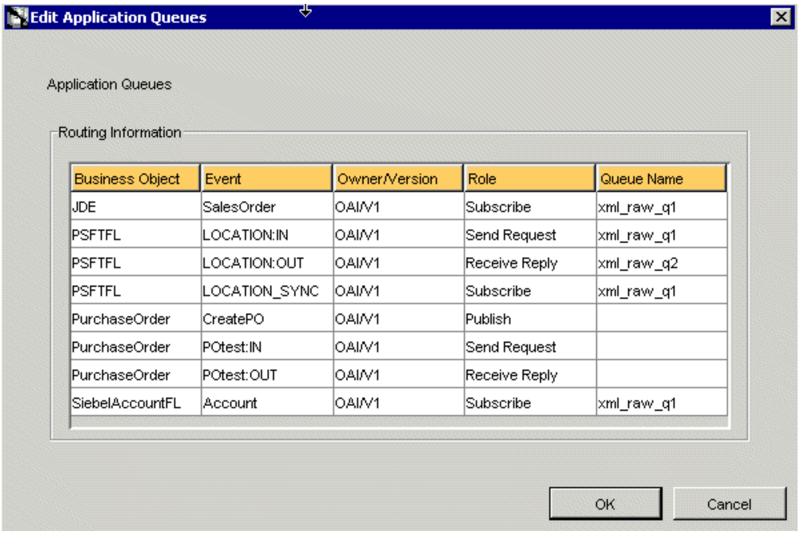
   You have completed defining the application definition for the invoked procedure.

Defining Queues for Advanced Queuing (AQ)

You must create and specify a queue for Advanced Queuing (AQ):



1. To specify the queue in iStudio, select the **Deploy** tab, **Project**, **Application**, **AQAPP**, **Routing**, and **Application Queues**.
2. Right-click **Application Queues** and select **Edit**.



Application Queues

Routing Information

Business Object	Event	Owner/Version	Role	Queue Name
JDE	SalesOrder	OAI/V1	Subscribe	xml_raw_q1
PSFTFL	LOCATION:IN	OAI/V1	Send Request	xml_raw_q1
PSFTFL	LOCATION:OUT	OAI/V1	Receive Reply	xml_raw_q2
PSFTFL	LOCATION_SYNC	OAI/V1	Subscribe	xml_raw_q1
PurchaseOrder	CreatePO	OAI/V1	Publish	
PurchaseOrder	POtest:IN	OAI/V1	Send Request	
PurchaseOrder	POtest:OUT	OAI/V1	Receive Reply	
SiebelAccountFL	Account	OAI/V1	Subscribe	xml_raw_q1

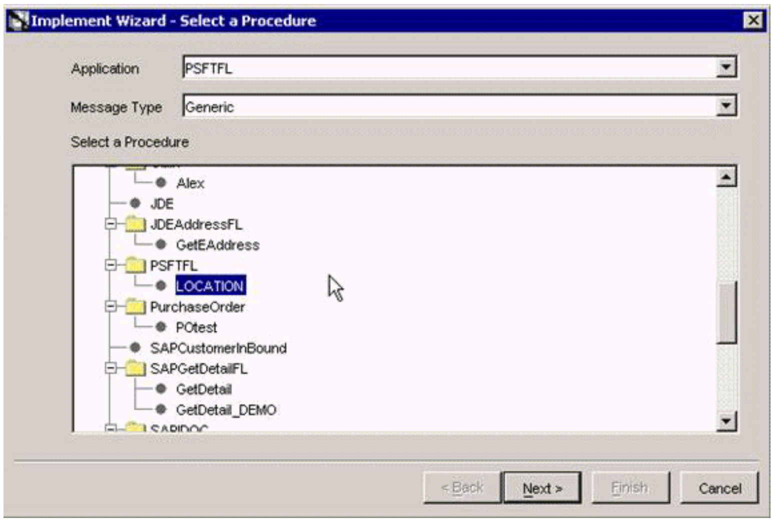
3. In the **Queue Name** column, type the queue name, in this example, **xml_raw_ql** for `PSFTFL LOCATION:IN` event, and **xml_raw_q2** for `PSFTFL LOCATION:OUT` event.
4. To finish, click **OK**.
5. To create xml_raw_ql and xml_raw_q2 queues, execute the following SQL under the appropriate schema, in this example, the AQAPP schema.

```
EXECUTE dbms_aqadm.create_queue_table (queue_table => 'RawMsgs_qtab', queue_
payload_type => 'RAW', multiple_consumers => FALSE);
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q1', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q1');
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q2', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q2');
```

Implementing the Procedure

To implement the procedure:

1. Create a new application called `PSFTFL`, expand the application, right-click **Implemented Procedure**, and select **New**.



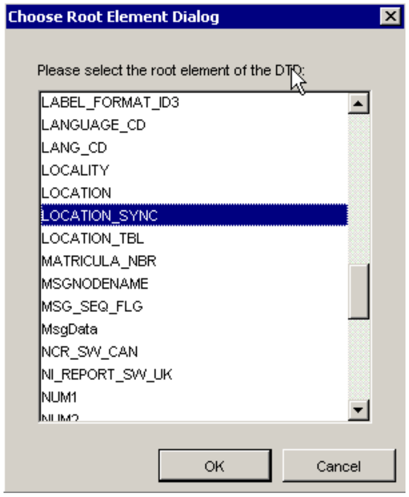
Implement Wizard - Select a Procedure

Application: PSFTFL

Message Type: Generic

2. From the **Message Type** list, select **Generic**.
3. Select **LOCATION** under **PSFTFL** as the procedure.
4. Click **Next**.
5. Open the DTD generated from Application Explorer and load it.

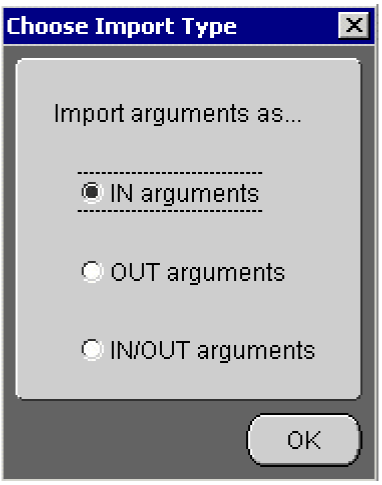
   The Choose Root Element dialog box opens.



Choose Root Element Dialog

Please select the root element of the DTD:

6. Select the root element, `PS8`, for this example.
7. Click **OK**.



Choose Import Type

Import arguments as...

● IN arguments

○ OUT arguments

○ IN/OUT arguments

OK

8. Select **IN arguments** as the import type for the request DTD and click **OK**.
9. In the **Name** field, type the root element of the request DTD if it is not populated automatically after importing the request DTD.

Editing the adapter.ini File

To edit the adapter.ini file:

1. Open the `adapter.ini` file.
2. Add the following two lines to `adapter.ini` for the adapter:

```
// Bridge class
bridge_class=com.iwaysoftware.iwbridge.IWBridge
// IBSE URL
ibse_url=http://hostname:port/ibse/IBSEServlet/XDSOAPRouter
```

   Where `hostname` is the name of the Oracle Application Server host and `port` is the server port number.

Runtime Configuration

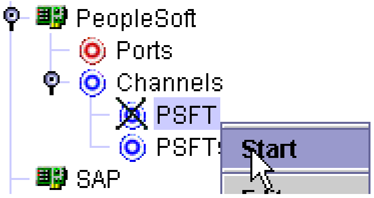
Triggering an Event in PeopleSoft

To trigger an event in your PeopleSoft system and send the event to the BSE listener:

1. Start Oracle Application Server or verify that it is running. Restart OC4J if necessary by issuing the following command:

```
OracleAS_HOME\opmn\bin\opmnctl  stopproc process-type=home
OracleAS_HOME\opmn\bin\opmnctl  startproc process-type=home
```

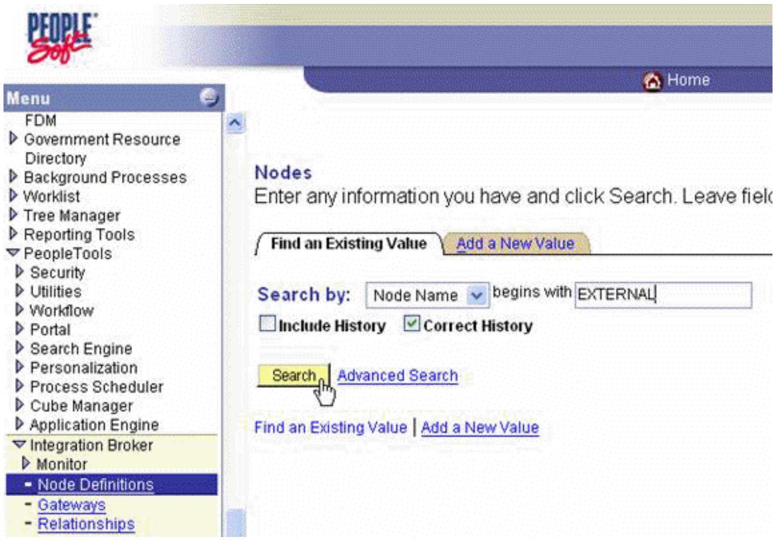
2. Start the channel in Application Explorer:



3. Start both the publishing adapter and the subscribing adapter by issuing the following commands:

```
InterConnect_home\adapters\AQAPP\start.bat
InterConnect_home\adapters\PSFTFL\start.bat
```

4. Log in to PeopleSoft, select **PeopleTools**, **Integration Broker**, and then **Node Definitions**.



Nodes

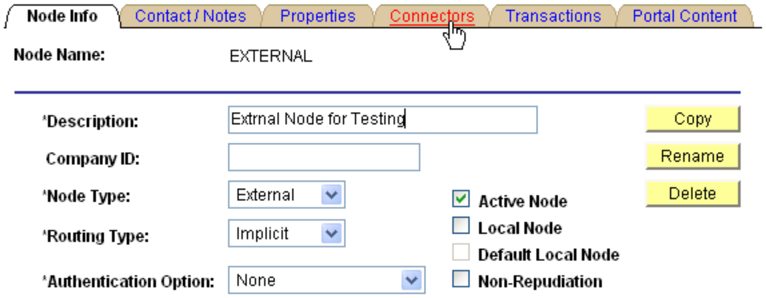
Enter any information you have and click Search. Leave fiel

Find an Existing Value | Add a New Value

Search by: Node Name begins with EXTERNAL

□ Include History ☑ Correct History

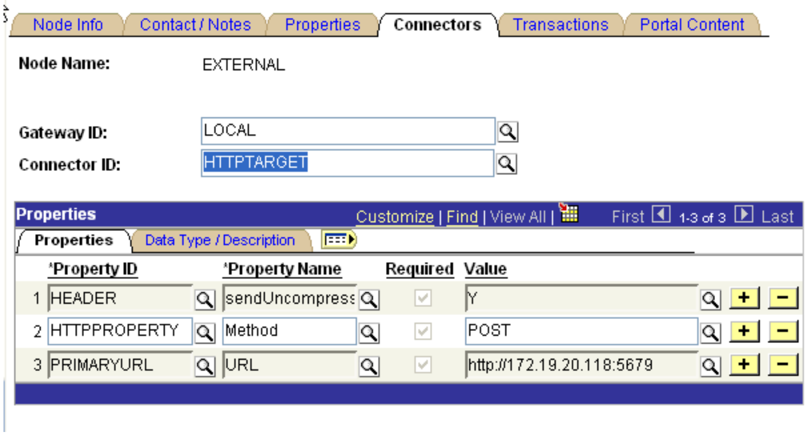
5. Type the node name you will be using, for example, EXTERNAL; then click **Search**.



Node Name:	EXTERNAL
*Description:	Extrnal Node for Testing
Company ID:	
*Node Type:	External
*Routing Type:	Implicit
*Authentication Option:	None

☑ Active Node
□ Local Node
□ Default Local Node
□ Non-Repudiation

6. Click the **Connectors** tab.



Node Name: EXTERNAL

Gateway ID: LOCAL

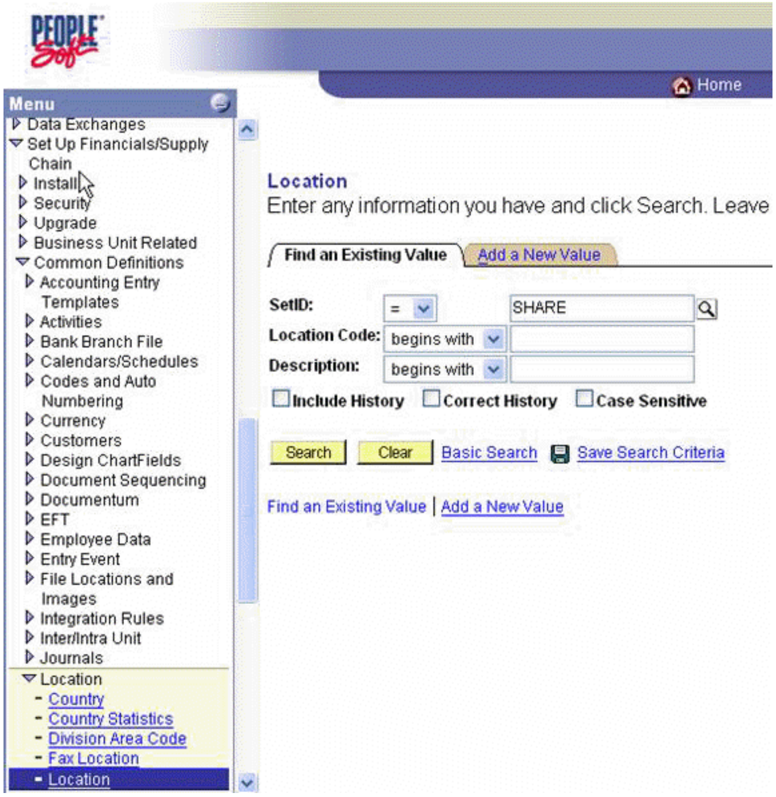
Connector ID: HTTPTARGET

Properties

	*Property ID	*Property Name	Required	Value
1	HEADER	sendUncompress	✓	Y
2	HTTPPROPERTY	Method	✓	POST
3	PRIMARYURL	URL	✓	http://172.19.20.118:5679

7. Enter **HTTPTARGET** in the **Connector ID** field; then enter the **iBSE listener URL** and its port in the **PRIMARYURL** field.

8. To save the configuration, click **Save**.



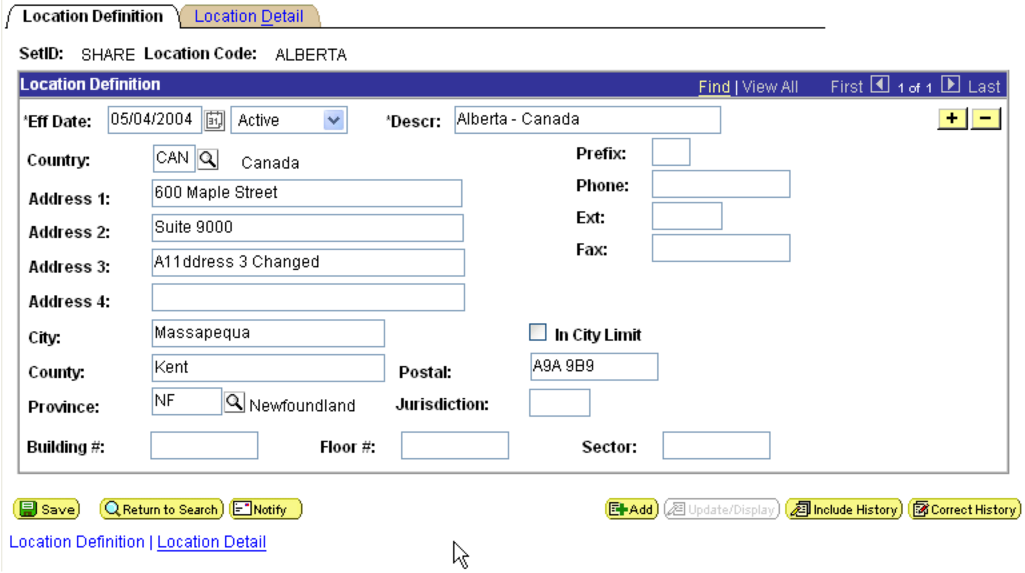
Location

Enter any information you have and click Search. Leave

SetID:	=	SHARE
Location Code:	begins with	
Description:	begins with	

□ Include History □ Correct History □ Case Sensitive

9. From **Menu** in the left pane, select **Set Up Financials/Supply Chain**, **Common Definitions**, **Location**, and then **Location**.



SetID: SHARE Location Code: ALBERTA

Location Definition

*Eff Date:	05/04/2004 Active	*Descr:	Alberta - Canada
Country:	CAN Canada	Prefix:	
Address 1:	600 Maple Street	Phone:	
Address 2:	Suite 9000	Ext:	
Address 3:	A11ddress 3 Changed	Fax:	
Address 4:			
City:	Massapequa	□ In City Limit	
County:	Kent	Postal:	A9A 9B9
Province:	NF Newfoundland	Jurisdiction:	
Building #:		Floor #:	
Sector:			

10. To find the location record you want to update, click **Search**; then make the changes and click **Save**.

When the change is saved, it triggers an event and sends the event to the BSE listener.

Verifying Results

The following code is an example of the PSFTFL log file:

```
Tue Nov 09 20:45:18 GMT-05:00 2004: The message was sent to topic(s) {oai_hub_
queue=[AQAPP]}. Processing Time = 10,203 ms.
<?xml version = '1.0' encoding = 'UTF-8'?>
<!DOCTYPE MSG>
<MSG>
  <H>
     <BO>PSFTFL</BO>
     <EN>LOCATION_SYNC</EN>
     <EV>OAI/V1</EV>
     <MV>OAI/V1</MV>
     <T>0</T>
     <SN>PSFTFL</SN>
     <SA>PSFTFL</SA>
     <SAID>41</SAID>
     <CI>PSFTFL1100051108265</CI>
  </H>
  <B>
    <AO N = "LOCATION_SYNC_CO">
      <AO N = "LOCATION_SYNC">
        <AO N = "FieldTypes">
          <AO N = "LOCATION_TBL">
            <A N = "class">R</A>
            <AO N = "SETID">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "LOCATION">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "EFFDT">
              <A N = "type">DATE</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "EFF_STATUS">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "DESCR">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "DESCR_AC">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "DESCRSHORT">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "BUILDING">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
```

```
<AO N = "FLOOR">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "SECTOR">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "JURISDICTION">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ATTN_TO">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "COUNTRY">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS1">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS2">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS3">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS4">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "CITY">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NUM1">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NUM2">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "HOUSE_TYPE">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD1">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD2">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
```

```
           <AO N = "ADDR_FIELD3">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "COUNTY">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "STATE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "POSTAL">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "GEO_CODE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "IN_CITY_LIMIT">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "COUNTRY_CODE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "PHONE">
Transform: CopyFields
            <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "EXTENSION">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "FAX">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "SETID_SALARY">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "SAL_ADMIN_PLAN">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "LANG_CD">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "HOLIDAY_SCHEDULE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "ESTABID">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
```

```
</AO>
<AO N = "LOCALITY">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "CAN_CMA">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "CAN_OEE_AREACD">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "GEOLOC_CODE">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "OFFICE_TYPE">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NCR_SW_CAN">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "TBS_OFFICE_CD_CAN">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "WRKS_CNCL_ID_LCL">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "SPK_COMM_ID_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_AREA_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "INDUST_INSP_ID_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NI_REPORT_SW_UK">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_GEOLOC_CD">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_DESIG_AGENT">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
  </AO>
  <AO N = "SOC_SEC_WRK_CTR">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "MATRICULA_NBR">
    <A N = "type">NUMBER</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "LABEL_FORMAT_ID2">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "LABEL_FORMAT_ID3">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "USG_LBL_FORMAT_ID">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "COMMENTS_2000">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "REG_REGION">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
</AO>
<AO N = "PSCAMA">
  <A N = "class">R</A>
  <AO N = "LANGUAGE_CD">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "AUDIT_ACTN">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "BASE_LANGUAGE_CD">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "MSG_SEQ_FLG">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "PROCESS_INSTANCE">
    <A N = "type">NUMBER</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "PUBLISH_RULE_ID">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  <AO N = "MSGNODENAME">
    <A N = "type">CHAR</A>
    <A N = "PCDATA"></A>
  </AO>
  </AO>
</AO>
<AO N = "MsgData">
  <AO N = "Transaction">
    <AO N = "LOCATION_TBL">
      <A N = "class">R</A>
      <AO N = "SETID">
        <A N = "PCDATA">SHARE</A>
      </AO>
      <AO N = "LOCATION">
        <A N = "PCDATA">ACT001x</A>
      </AO>
      <AO N = "EFFDT">
        <A N = "PCDATA">2002-07-26</A>
      </AO>
      <AO N = "EFF_STATUS">
        <A N = "PCDATA">A</A>
      </AO>
      <AO N = "DESCR">
        <A N = "PCDATA">Testing PSoft Runtime - 2</A>
      </AO>
      <AO N = "DESCR_AC">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "DESCRSHORT">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "BUILDING">
        <A N = "PCDATA">2</A>
      </AO>
      <AO N = "FLOOR">
        <A N = "PCDATA">29</A>
      </AO>
      <AO N = "SECTOR">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "JURISDICTION">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "ATTN_TO">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "COUNTRY">
        <A N = "PCDATA">USA</A>
      </AO>
      <AO N = "ADDRESS1">
        <A N = "PCDATA">2 Pennn Plazaa</A>
      </AO>
      <AO N = "ADDRESS2">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "ADDRESS3">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "ADDRESS4">
        <A N = "PCDATA"></A>
      </AO>
      <AO N = "CITY">
        <A N = "PCDATA">New York</A>
      </AO>
```

```
<AO N = "NUM1">
  <A N = "PCDATA"></A>
</AO>
<AO N = "NUM2">
  <A N = "PCDATA"></A>
</AO>
<AO N = "HOUSE_TYPE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD1">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD2">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD3">
  <A N = "PCDATA"></A>
</AO>
<AO N = "COUNTY">
  <A N = "PCDATA"></A>
</AO>
<AO N = "STATE">
  <A N = "PCDATA">NY</A>
</AO>
<AO N = "POSTAL">
  <A N = "PCDATA">10121</A>
</AO>
<AO N = "GEO_CODE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "IN_CITY_LIMIT">
  <A N = "PCDATA"></A>
</AO>
<AO N = "COUNTRY_CODE">
  <A N = "PCDATA">212</A>
</AO>
<AO N = "PHONE">
  <A N = "PCDATA">736-4433</A>
</AO>
<AO N = "EXTENSION">
  <A N = "PCDATA">3694</A>
</AO>
<AO N = "FAX">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SETID_SALARY">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SAL_ADMIN_PLAN">
  <A N = "PCDATA"></A>
</AO>
<AO N = "LANG_CD">
  <A N = "PCDATA"></A>
</AO>
<AO N = "HOLIDAY_SCHEDULE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ESTABID">
  <A N = "PCDATA"></A>
</AO>
```

```
<AO N = "LOCALITY">
  <A N = "PCDATA"></A>
</AO>
<AO N = "CAN_CMA">
  <A N = "PCDATA"></A>
</AO>
<AO N = "CAN_OEE_AREACD">
  <A N = "PCDATA"></A>
</AO>
<AO N = "GEOLOC_CODE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "OFFICE_TYPE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "NCR_SW_CAN">
  <A N = "PCDATA"></A>
</AO>
<AO N = "TBS_OFFICE_CD_CAN">
  <A N = "PCDATA"></A>
</AO>
<AO N = "WRKS_CNCL_ID_LCL">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SPK_COMM_ID_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_AREA_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "INDUST_INSP_ID_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "NI_REPORT_SW_UK">
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_GEOLOC_CD">
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_DESIG_AGENT">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SOC_SEC_WRK_CTR">
  <A N = "PCDATA"></A>
</AO>
<AO N = "MATRICULA_NBR">
  <A N = "PCDATA">0</A>
</AO>
<AO N = "LABEL_FORMAT_ID2">
  <A N = "PCDATA"></A>
</AO>
<AO N = "LABEL_FORMAT_ID3">
  <A N = "PCDATA"></A>
</AO>
<AO N = "USG_LBL_FORMAT_ID">
  <A N = "PCDATA"></A>
</AO>
```

```
              <AO N = "COMMENTS_2000">
                <A N = "PCDATA"></A>
              </AO>
              <AO N = "REG_REGION">
                <A N = "PCDATA"></A>
              </AO>
            </AO>
          </AO>
        </AO>
      </AO>
    </AO>
  </B>
</MSG>
Tue Nov 09 20:45:18 GMT-05:00 2004: Outbound Transform Engine: beginning to
transform message.
Tue Nov 09 20:45:18 GMT-05:00 2004: Outbound Transform Engine: performing
transform.
```

Subscribing Adapter (AQAPP) Console Log

```
Tue Nov 09 20:45:11 GMT-05:00 2004: AQ Adapter: received the message from the
Agent and will now write it to AQ.
Tue Nov 09 20:45:12 GMT-05:00 2004: AQ Adapter: successfully converted the OAI
message to XML
<?xml version = '1.0' encoding = 'UTF-8' standalone = 'yes'?>
<LOCATION_SYNC>
   <FieldTypes>
      <LOCATION_TBL class="R">
         <SETID type="CHAR"></SETID>
         <LOCATION type="CHAR"></LOCATION>
         <EFFDT type="DATE"></EFFDT>
         <EFF_STATUS type="CHAR"></EFF_STATUS>
         <DESCR type="CHAR"></DESCR>
         <DESCR_AC type="CHAR"></DESCR_AC>
         <DESCRSHORT type="CHAR"></DESCRSHORT>
         <BUILDING type="CHAR"></BUILDING>
         <FLOOR type="CHAR"></FLOOR>
         <SECTOR type="CHAR"></SECTOR>
         <JURISDICTION type="CHAR"></JURISDICTION>
         <ATTN_TO type="CHAR"></ATTN_TO>
         <COUNTRY type="CHAR"></COUNTRY>
         <ADDRESS1 type="CHAR"></ADDRESS1>
         <ADDRESS2 type="CHAR"></ADDRESS2>
         <ADDRESS3 type="CHAR"></ADDRESS3>
         <ADDRESS4 type="CHAR"></ADDRESS4>
         <CITY type="CHAR"></CITY>
         <NUM1 type="CHAR"></NUM1>
         <NUM2 type="CHAR"></NUM2>
         <HOUSE_TYPE type="CHAR"></HOUSE_TYPE>
         <ADDR_FIELD1 type="CHAR"></ADDR_FIELD1>
         <ADDR_FIELD2 type="CHAR"></ADDR_FIELD2>
         <ADDR_FIELD3 type="CHAR"></ADDR_FIELD3>
         <COUNTY type="CHAR"></COUNTY>
         <STATE type="CHAR"></STATE>
         <POSTAL type="CHAR"></POSTAL>
         <GEO_CODE type="CHAR"></GEO_CODE>
         <IN_CITY_LIMIT type="CHAR"></IN_CITY_LIMIT>
         <COUNTRY_CODE type="CHAR"></COUNTRY_CODE>
         <PHONE type="CHAR"></PHONE>
         <EXTENSION type="CHAR"></EXTENSION>
         <FAX type="CHAR"></FAX>
      <SETID_SALARY type="CHAR"></SETID_SALARY>
      <SAL_ADMIN_PLAN type="CHAR"></SAL_ADMIN_PLAN>
      <LANG_CD type="CHAR"></LANG_CD>
      <HOLIDAY_SCHEDULE type="CHAR"></HOLIDAY_SCHEDULE>
      <ESTABID type="CHAR"></ESTABID>
      <LOCALITY type="CHAR"></LOCALITY>
      <CAN_CMA type="CHAR"></CAN_CMA>
      <CAN_OEE_AREACD type="CHAR"></CAN_OEE_AREACD>
      <GEOLOC_CODE type="CHAR"></GEOLOC_CODE>
      <OFFICE_TYPE type="CHAR"></OFFICE_TYPE>
      <NCR_SW_CAN type="CHAR"></NCR_SW_CAN>
      <TBS_OFFICE_CD_CAN type="CHAR"></TBS_OFFICE_CD_CAN>
      <WRKS_CNCL_ID_LCL type="CHAR"></WRKS_CNCL_ID_LCL>
      <SPK_COMM_ID_GER type="CHAR"></SPK_COMM_ID_GER>
      <TARIFF_AREA_GER type="CHAR"></TARIFF_AREA_GER>
      <TARIFF_GER type="CHAR"></TARIFF_GER>
      <INDUST_INSP_ID_GER type="CHAR"></INDUST_INSP_ID_GER>
      <NI_REPORT_SW_UK type="CHAR"></NI_REPORT_SW_UK>
      <GVT_GEOLOC_CD type="CHAR"></GVT_GEOLOC_CD>
      <GVT_DESIG_AGENT type="CHAR"></GVT_DESIG_AGENT>
      <SOC_SEC_WRK_CTR type="CHAR"></SOC_SEC_WRK_CTR>
      <MATRICULA_NBR type="NUMBER"></MATRICULA_NBR>
      <LABEL_FORMAT_ID2 type="CHAR"></LABEL_FORMAT_ID2>
      <LABEL_FORMAT_ID3 type="CHAR"></LABEL_FORMAT_ID3>
      <USG_LBL_FORMAT_ID type="CHAR"></USG_LBL_FORMAT_ID>
      <COMMENTS_2000 type="CHAR"></COMMENTS_2000>
      <REG_REGION type="CHAR"></REG_REGION>
   </LOCATION_TBL>
   <PSCAMA class="R">
      <LANGUAGE_CD type="CHAR"></LANGUAGE_CD>
      <AUDIT_ACTN type="CHAR"></AUDIT_ACTN>
      <BASE_LANGUAGE_CD type="CHAR"></BASE_LANGUAGE_CD>
      <MSG_SEQ_FLG type="CHAR"></MSG_SEQ_FLG>
      <PROCESS_INSTANCE type="NUMBER"></PROCESS_INSTANCE>
      <PUBLISH_RULE_ID type="CHAR"></PUBLISH_RULE_ID>
      <MSGNODENAME type="CHAR"></MSGNODENAME>
   </PSCAMA>
</FieldTypes>
<MsgData>
   <Transaction>
      <LOCATION_TBL class="R">
         <SETID>SHARE</SETID>
         <LOCATION>ACT001x</LOCATION>
         <EFFDT>2002-07-26</EFFDT>
         <EFF_STATUS>A</EFF_STATUS>
         <DESCR>Testing PSoft Runtime - 2</DESCR>
         <DESCR_AC></DESCR_AC>
         <DESCRSHORT></DESCRSHORT>
         <BUILDING>2</BUILDING>
         <FLOOR>29</FLOOR>
         <SECTOR></SECTOR>
         <JURISDICTION></JURISDICTION>
         <ATTN_TO></ATTN_TO>
         <COUNTRY>USA</COUNTRY>
         <ADDRESS1>2 Pennn Plazaa</ADDRESS1>
         <ADDRESS2></ADDRESS2>
         <ADDRESS3></ADDRESS3>
         <ADDRESS4></ADDRESS4>
         <CITY>New York</CITY>
         <NUM1></NUM1>
```

```
            <NUM2></NUM2>
            <HOUSE_TYPE></HOUSE_TYPE>
            <ADDR_FIELD1></ADDR_FIELD1>
            <ADDR_FIELD2></ADDR_FIELD2>
            <ADDR_FIELD3></ADDR_FIELD3>
            <COUNTY></COUNTY>
            <STATE>NY</STATE>
            <POSTAL>10121</POSTAL>
            <GEO_CODE></GEO_CODE>
            <IN_CITY_LIMIT></IN_CITY_LIMIT>
            <COUNTRY_CODE>212</COUNTRY_CODE>
            <PHONE>736-4433</PHONE>
            <EXTENSION>3694</EXTENSION>
            <FAX></FAX>
            <SETID_SALARY></SETID_SALARY>
            <SAL_ADMIN_PLAN></SAL_ADMIN_PLAN>
            <LANG_CD></LANG_CD>
            <HOLIDAY_SCHEDULE></HOLIDAY_SCHEDULE>
            <ESTABID></ESTABID>
            <LOCALITY></LOCALITY>
            <CAN_CMA></CAN_CMA>
            <CAN_OEE_AREACD></CAN_OEE_AREACD>
            <GEOLOC_CODE></GEOLOC_CODE>
            <OFFICE_TYPE></OFFICE_TYPE>
            <NCR_SW_CAN></NCR_SW_CAN>
            <TBS_OFFICE_CD_CAN></TBS_OFFICE_CD_CAN>
            <WRKS_CNCL_ID_LCL></WRKS_CNCL_ID_LCL>
            <SPK_COMM_ID_GER></SPK_COMM_ID_GER>
            <TARIFF_AREA_GER></TARIFF_AREA_GER>
            <TARIFF_GER></TARIFF_GER>
            <INDUST_INSP_ID_GER></INDUST_INSP_ID_GER>
            <NI_REPORT_SW_UK></NI_REPORT_SW_UK>
            <GVT_GEOLOC_CD></GVT_GEOLOC_CD>
            <GVT_DESIG_AGENT></GVT_DESIG_AGENT>
            <SOC_SEC_WRK_CTR></SOC_SEC_WRK_CTR>
            <MATRICULA_NBR>0</MATRICULA_NBR>
            <LABEL_FORMAT_ID2></LABEL_FORMAT_ID2>
            <LABEL_FORMAT_ID3></LABEL_FORMAT_ID3>
            <USG_LBL_FORMAT_ID></USG_LBL_FORMAT_ID>
            <COMMENTS_2000></COMMENTS_2000>
            <REG_REGION></REG_REGION>
         </LOCATION_TBL>
      </Transaction>
   </MsgData>
</LOCATION_SYNC>
```

7

Troubleshooting and Error Messages

This chapter explains the limitations and workarounds when connecting to PeopleSoft. The following topics are discussed:

- Troubleshooting
- BSE Error Messages

The adapter-specific errors listed in this chapter can arise whether using the adapter with an OracleAS Adapter J2CA or with an OracleAS Adapter Business Services Engine (BSE) configuration.

Troubleshooting

This topic provides troubleshooting information for PeopleSoft, separated into four categories:

- Application Explorer
- PeopleSoft
- OracleAS Adapter J2CA
- BSE

> **Note:** Log file information that can be relevant in troubleshooting can be found in the following locations:
>
> - The OracleAS Adapter J2CA trace information can be found under the `*OracleAS_home*\opmn\logs` directory.
> - BSE trace information can be found under the `*OracleAS_home*\j2ee\home\applications\ws-app-adapter\ibse\ibselogs` directory.
> - The log file for Application Explorer can be found under the `*OracleAS_home*\adapters\application\tools` directory.

Application Explorer

To use Application Explorer on **Windows** for debugging or testing purposes, load the batch script `ae.bat`, found under:

```
OracleAS_home\adapters\application\tools
```

On **UNIX**, load the shell script `ae.sh`, found under:

```
OracleAS_home/adapters/application/tools
```

Error	Solution
Cannot connect to OracleAS Adapter for PeopleSoft from Application Explorer. The following error message appears: `Problem activating adapter`	Ensure that: ■ PeopleSoft is running. ■ The PeopleSoft user ID and password are correct. ■ The port number is correct. ■ The custom component interface is properly installed.
The following error message appears: java.lang.IllegalStateException: java.lang.Exception: Error Logon to PeopleSoft System	You have provided invalid connection information for PeopleSoft or the wrong `psjoa.jar` is in the lib directory. The `psjoa.jar` file version is specific to the PeopleTools release.
PeopleSoft does not appear in the Application Explorer Adapter node list.	Ensure that the PeopleSoft JAR files, `iwpsci84.jar` (or iwpsci81.jar) and `psjoa.jar,` are added to the `lib` directory.
Logon failure error at runtime.	If the password for connecting to your PeopleSoft system is not specified when creating a target or with the Edit option in Application Explorer, you will be unable to connect to PeopleSoft. The connection password is not saved in `repository.xml`. Update the password using the Edit option in Application Explorer, then restart the application server.
The following error message appears: Jolt Session Pool cannot provide a connection to the appserver. This appears to be because there is no available application server domain. [Fri Aug 27 13:06:27 EDT 2004] bea.jolt.ServiceException: Invalid Session	The host name or port number for PeopleSoft is incorrect.
Properties are not displayed for a component interface.	You are using the wrong `iwpsci8x.jar` file.
Cannot generate schemas.	If the error message “Index: -1, Size:0” appears, or if you can log on to Application Explorer but you cannot see any Component Interfaces or Messages, then you may have both the `iwpsci81.jar` and `iwpsci84.jar` files in your `lib` directory. Stop your server, remove the unrequired jar file, and restart the server.
The following exception occurs when you start Application Explorer by activating `ae.bat` (not `iaexplorer.exe`): `java.lang.ClassNotFoundException: org.bouncycastle.jce.provider.BouncyCastleProvider`	This is a benign exception. It does not affect adapter functionality. Download BouncyCastle files from: `ftp://ftp.bouncycastle.org/pub`

Error	Solution

Unable to start Application Explorer in a Solaris environment. The following exception is thrown in the console:
`javax.resource.ResourceException: IWAFManagedConnectionFactory: License violation.at com.ibi.afjca.spi.IWAFManagedConnectionFactory.createConnectionFactory(IWAFManagedConnectionFactory.java:98)at com.iwaysoftware.iwae.common.JCATransport.getConnectionFactory(JCATransport.java:133) at com.iwaysoftware.iwae.common.JCATransport.initJCA(JCATransport.java:69)at com.iwaysoftware.iwae.common.JCATransport.<init>(JCATransport.java:62)at com.iwaysoftware.iwae.common.AdapterClient.<init>(AdapterClient.java:85)at com.ibi.bse.ConfigWorker.run(ConfigWorker.java:41)at java.lang.Thread.run(Thread.java:534)`
`Could not create the connection factory.`
	`JAVACMD` is not set on the user system. Before starting Application Explorer, export `JAVACMD` as follows:
	`JAVACMD=/<jdk_home>/bin/java`, where `<jdk_home>` is the directory where JDK is installed on your machine.

PeopleSoft

Error	Solution
Services are not working properly when using the PeopleSoft Component Interface testing tool in three-tier mode.	To test properly using the Component Interface testing tool: **1.** Open Application Designer. **2.** Select the Component Interface. **3.** Use the test tool. If service works in test tool, then review the XML and check for redundant fields in XML.
The following error message appears: `Jolt Session Pool cannot provide a connection to the appserver. This appears to be because there is no available application server domain. [Fri Aug 27 13:06:27 EDT 2004] bea.jolt.ServiceException: Invalid Session`	The host name or port number for PeopleSoft is incorrect.
Component Interfaces and Messages do not appear in the adapter tree.	The project is not installed properly on the PeopleSoft system.

Error	Solution

Return error code `-1` is received from PeopleSoft at runtime, for example:
`<LOCATIONProcessResponse xmlns="http://xmlns.oracle.com/LOCATION">`
`<error xmlns="">-1</error>`
`</LOCATIONProcessResponse>`
	You are either using the incorrect version of `psjoa.jar`, or have both the `iwpsci81.jar` and `iwpsci84.jar` files in your `lib` directory. In the second case, you must delete the unused JAR file, and then restart the server.
	The `psjoa.jar` file version is specific to the PeopleTools release.

Pstools.properties file has not been initialized.	This file is required for PeopleSoft 8.1. If you are using PeopleSoft 8.1, you should add this file. If you are not using PeopleSoft 8.1 and this error message still appears, ignore the message.
The following error message appears: `Cannot find Component Interface {CI name}`	The reason may be either of the following: ■ The Java API for the selected component interface is not found in the API JAR file. Please check the Java API for the class file for the CI. If not found, please add the class file for the CI. ■ The component interface name is mentioned incorrectly in the request document.
The following error message appears: `Not Authorized (90,6) Failed to execute PSSession request`	The component interface does not have the necessary access to perform the operation. Change the permission settings in the PeopleSoft > Security > Permission list for the component interface.
The following error message appears: `Must also provide values for keys {keyname}`	The reason may be any of the following: ■ The request XML document does not have the element for the mandatory key. Please include the keyname and the value in the request document. ■ The Key field name is mentioned incorrectly in the request document. ■ The Perform operation is mentioned incorrectly in the request XML document.

OracleAS Adapter J2CA

Error	Solution
In Application Explorer, the following error message appears when you attempt to connect to an OracleAS Adapter J2CA configuration: `Could not initialize JCA`	In the Details tab in the right pane, ensure that the directory specified in the Home field points to the correct directory, for example, *`OracleAS_home`*`\adapters\application`

BPEL Process Manager

Error	Solution
Endpoint activation error on deployment of PeopleSoft event handling project (inbound) in JDeveloper	Verify that the channel used for this inbound J2CA service is stopped in Application Explorer. If you have started this channel for testing or debugging purposes, you must stop it before starting BPEL PM Server. Endpoint activation is managed by BPEL Process Manager.
The following error message appears in BPEL PM Server Console: `Process "TestPSFT" (revision "1.0") compilation failed. <2005-05-18 10:49:53,285> <ERROR><default.collaxa.cube.engine.deployment> <Cube ProcessLoader::create> Failed to read wsdl. Error happened when reading wsdl at "http://127.0.0.1:7777/BPELConsole/wsil/adapters/applications/CUSTOMER_invoke.wsdl?wsdl", because "WSDLException: faultCode=INVALID_WSDL: Invalid XML in document at: http://127.0.0.1:7777/BPELConsole/wsil/adapters/applications/CUSTOMER_invoke.wsdl?wsdl: The element type "P" must be terminated by the matching end-tag "</P>".`	Verify that the specified WSDL file exists at that URL and that the file is valid. Workaround: Change the WSDL location to `localhost:7777`. The default is `127.0.0.1:7777`. Alternative workaround: Add the IP address to the `Dhttp.nonProxyHosts` list found in `obsetenv.bat` (Windows) or `obsetenv.sh` (Unix)
The following exception is thrown in JDeveloper during deployment of the BPEL process: `java.io.FileNotFoundException: \BPELConsole\wsil\adapters\applications\LOCATION_SYNC.VERSION_1_receive.wsdl?wsdl (The system cannot find the path specified)`	Verify that you have all the required patches installed. The required patches are listed and updated on the Oracle Technology Network Web site

BSE Error Messages

This topic discusses the different types of errors that can occur when processing Web services through Oracle Application Server Adapter Business Services Engine (BSE).

General Error Handling in BSE

BSE serves as both a SOAP gateway into the adapter framework and as the engine for some of the adapters. In both design time and runtime, various conditions can cause errors in BSE when Web services that use adapters are running. Some of these conditions and resulting errors are exposed the same way, regardless of the specific adapter; others are exposed differently, based on the adapter being used. This topic explains what you can expect when you encounter some of the more common error conditions on an adapter-specific basis.

Usually, the SOAP gateway (**agent**) inside BSE passes a SOAP request message to the adapter required for the Web service. If an error occurs, how it is exposed depends on the adapter and the API or interfaces that the adapter uses. The APIs are generated from PeopleTools and are specific to the PeopleTools release. A few scenarios cause the

SOAP gateway to generate a SOAP fault. In general, anytime the SOAP agent inside BSE receives an invalid SOAP request, a SOAP fault element is generated in the SOAP response. The SOAP fault element contains fault string and fault code elements. The fault code contains a description of the SOAP agent error.

The following SOAP response document results when BSE receives an invalid SOAP request:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">

 <SOAP-ENV:Body>
      <SOAP-ENV:Fault>
         <faultcode>SOAP-ENV:Client</faultcode>
         <faultstring>Parameter node is missing</faultstring>
      </SOAP-ENV:Fault>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

In this example, BSE did not receive an element in the SOAP request message that is mandatory for the WSDL for this Web service.

Adapter-Specific Error Handling

When an adapter raises an exception during runtime, the SOAP agent in BSE produces a SOAP fault element in the generated SOAP response. The SOAP fault element contains fault code and fault string elements. The fault string contains the native error description from the adapter target system. Since adapters use the target system interfaces and APIs, whether or not an exception is raised depends on how the target systems interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in BSE, and that request is invalid based on the WSDL for that service, the adapter may raise an exception yielding a SOAP fault.

While it is almost impossible to anticipate every error condition that an adapter may encounter, the following is a description of how adapters handle common error conditions and how they are then exposed to the Web services consumer application.

OracleAS Adapter for PeopleSoft Invalid SOAP Request

If the PeopleSoft agent receives a SOAP request message that does not conform to the WSDL for the Web service being executed, then the following SOAP response is generated.

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
  <SOAP-ENV:Body>
   <m:CARRIERResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
   xmlns="urn:schemas-iwaysoftware-com:iwse"
   cid="2A3CB42703EB20203F91951B89F3C5AF">
   <PS8>
      <error>Cannot find Component Interface {VARRIER}    (91,2)Initialization
    failed (90,7)Not Authorized (90,6)Failed to execute PSSession request Cannot
find Component Interface {VARRIER} (91,2)</error>
        </PS8>
      </m:CARRIERResponse>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Empty Result From PeopleSoft Request

If OracleAS Adapter for PeopleSoft executes a component interface as a Web service using input parameters passed in the SOAP request that do not match records in PeopleSoft, then the following SOAP response is generated.

```
<SOAP-ENV:Enve
lope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
 <SOAP-ENV:Body>
  <m:CARRIERResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
xmlns="urn:schemas-iwaysoftware-com:iwse" cid="2A3CB42703EB20203F91951B89F3C5AF">
    <PS8>
        <error>No rows exist for the specified keys. {CARRIER} (91,50)Failed to
execute PSBusComp request</error>
    </PS8>
  </m:CARRIERResponse>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Failure to Connect to PeopleSoft

If OracleAS Adapter for PeopleSoft cannot connect to PeopleSoft, then the following SOAP response is generated.

```
<?xml version="1.0" enco
ding="ISO-8859-1" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
   <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>java.lang.Exception: Error Logon to PeopleSoft
   System<faultstring>
 </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Invalid SOAP Request

If Oracle Application Server Adapter receives a SOAP request message that does not conform to the WSDL for the Web services being executed, then the following SOAP response is generated.

```
<?xml version="1.0" encoding="ISO-8859-1"
 ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
 <SOAP-ENV:Fault>
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>RPC server connection failed: Connection refused:
connect</faultstring>
 </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Empty Result From Oracle Application Server Adapter Request

If the adapter executes a SOAP request using input parameters passed that do not match records in the target system, then the following SOAP response is generated.

Note: The condition for this adapter does not yield a SOAP fault.

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
   <SOAP-ENV:Body>
      <m:RunDBQueryResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
        xmlns="urn:schemas-iwaysoftware-com:iwse"
        cid="2A3CB42703EB20203F91951B89F3C5AF">
        <RunDBQueryResult run="1" />
     </m:RunDBQueryResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

8

Advanced User Tools

This chapter includes the following topics:

- Web Services Policy-Based Security
- Migrating Repositories

Web Services Policy-Based Security

OracleAS Adapter Application Explorer (Application Explorer) provides a security model called Web services policy-based security. The following topics describe how the feature works and how to configure it.

Web services provide a layer of abstraction between the back-end business logic and the user or application running the Web service. This enables easy application integration but raises the issue of controlling the use and implementation of critical and sensitive business logic that is run as a Web service.

Application Explorer controls the use of Web services that use adapters, using a feature called policy-based security. This feature enables an administrator to apply policies to business services (Web services) to deny or permit their execution.

A policy is a set of privileges dealing with the execution of a business service that can be applied to an existing or new business service. When you set specific rights or privileges inside a policy, you do not have to re-create privileges for every business service that has security concerns in common with other business services. Instead, you reuse a policy on multiple business services.

The goal of the feature is to secure requests at both the transport and the SOAP request level transmitted on the wire. Some of the policies do not deal with security issues directly, but do affect the runtime behavior of the Web services to which they have been applied.

The BSE administrator creates an "instance" of a policy type, names it, associates individual users or groups (a collection of users), and then applies that policy to one or more business services.

You can assign a policy to a business service, or to a method within a business service. If a policy is only applied to a method, other methods in that business service will not be governed by it. However, if a policy is applied to the business service, all methods are governed by it. At runtime, the user ID and password that are sent to BSE in the SOAP request message are checked against the list of users for all policies applied to that specific business service. The policy type that is supported is Resource Execution, which dictates who can or cannot execute the business service.

When a policy is not applied, the default value for a business service is to "grant all". For example, anybody can execute the business service, until the Resource Execution policy is associated to the business service. At that time, only those granted execution permissions, or users not part of the group that has been denied execution permissions, have access to the business service.

Configuring Web Services Policy-Based Security

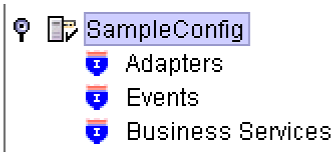
The following procedures describe how to configure Web services policy-based security.

Creating and Associating a User with a Policy

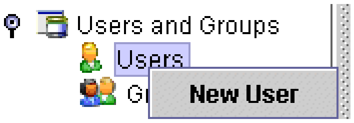
Before you create instances of policies, you must have a minimum of one user or one group to associate to an instance. You can create users and groups using Application Explorer.

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, `SampleConfig`. See Chapter 2, "Configuring OracleAS Adapter for PeopleSoft" for information on creating a new configuration.
3. Select **Connect**.

   Nodes appear for Adapters, Events, and Business Services (also known as Web services).

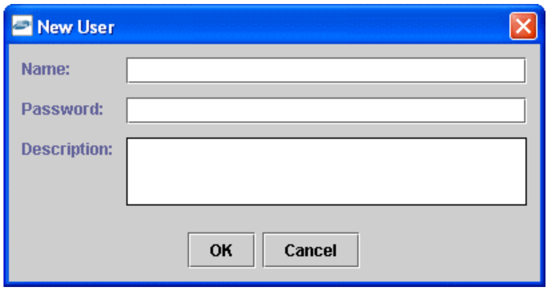


   a. Expand the **Business Services** node.
   b. Expand the **Configuration** node.
   c. Expand the **Security** node.
   d. Expand the **Users and Groups** node.



4. Right-click **Users** and click **New User**.

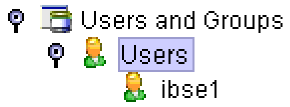
   The New User dialog box is displayed.



   a. In the **Name** field, enter a user ID.
   b. In the **Password** field, enter the password associated with the user ID.

c. In the **Description** field, enter a description of the user (optional).

5. Click **OK**.



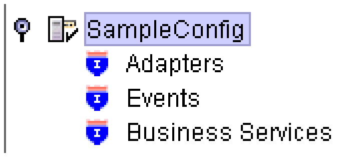
The new user is added under the Users node.

Creating a Group to Use With a Policy

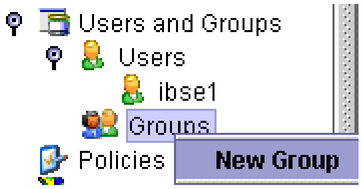
To create a group to use with a policy:

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Configuring OracleAS Adapter for PeopleSoft" for information on creating a new configuration.
3. Select **Connect**.

   Nodes appear for Adapters, Events, and Business Services (also known as Web services).

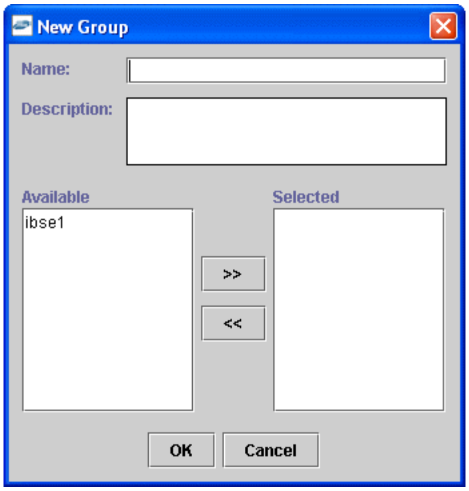


   a. Expand the **Business Services** node.
   b. Expand the **Configuration** node.
   c. Expand the **Security** node.
   d. Expand the **Users and Groups** node.



4. Right-click **Groups** and select **New Group**.

   The New Group dialog box is displayed.



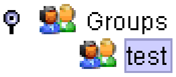
a. In the **Name** field, enter a name for the group.

b. In the **Description** field, enter a description for the group (optional).

c. From the available list of users in the left pane, select one or more users and add them to the **Selected** list by clicking the double right-facing arrow.

5. When you have selected at least one user, click **OK**.

The following shows the new group added under the **Groups** node.



Creating an Execution Policy

An execution policy governs who can execute the business services to which the policy is applied.

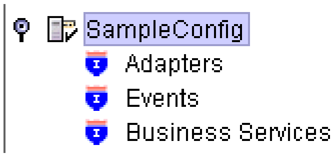
To create an execution policy:

1. Start Application Explorer.

2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Configuring OracleAS Adapter for PeopleSoft" for information on creating a new configuration.

3. Select **Connect**.

   Nodes appear for Adapters, Events, and Business Services (also known as Web services).

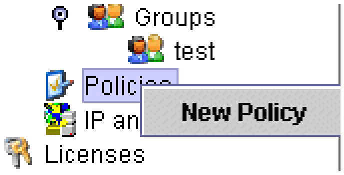


a. Expand the **Business Services** node.

b. Expand the **Configuration** node.

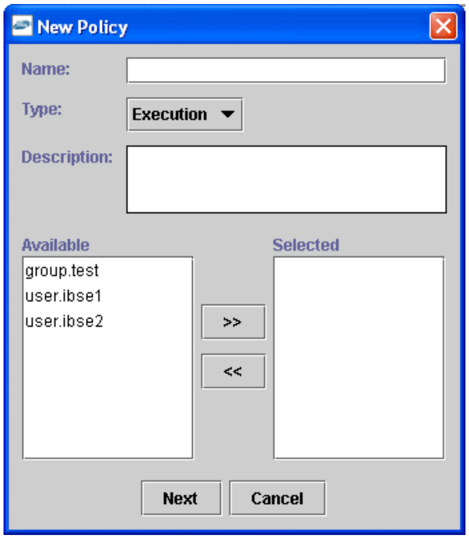
c. Expand the **Security** node.

d. Expand the **Policies** node.



4. Right-click **Policies** and select **New Policy**.

   The New policy dialog box is displayed.



   Provide the following information:

   a. In the **Name** field, enter a name for the policy.

   b. From the Type list, select **Execution**.

   c. In the **Description** field, enter a description for the policy (optional).

   d. From the available list of users in the left pane, select one or more users and add them to the **Selected** list by clicking the double right-facing arrow.

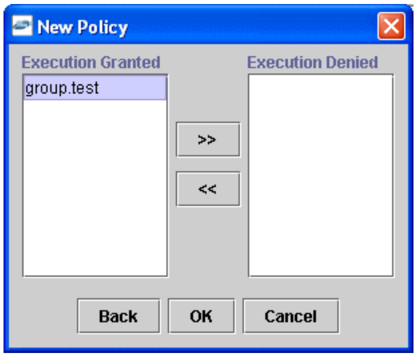
      Note: This user ID is checked against the value in the user ID element of the SOAP header sent to BSE in a SOAP request.

      ---

5. When you have selected at least one user, click **OK**.

6. Click **Next**.

The New Policy permissions dialog box is displayed.

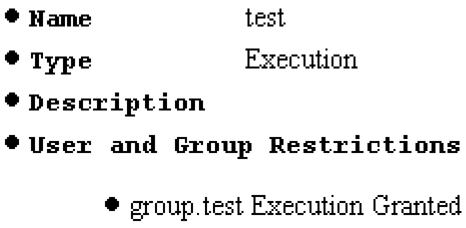


7. To grant permission to a user or group to execute a business service, select the user or group and move them into the **Execution Granted** list by selecting the double left-facing arrow.

8. To deny permission to a user or group to execute a business service, select the user or group and move them into the Execution Denied list by selecting the double right-facing arrow.

9. Click **OK**.

   The following pane summarizes your configuration.



- **Name** test
- **Type** Execution
- **Description**
- **User and Group Restrictions**
  - group.test Execution Granted

Using the IP and Domain Restrictions Policy Type

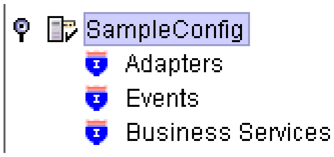
You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to BSE and therefore need not be applied to individual Web services. You need not create a policy; however, you must enable the Security Policy option in Application Explorer.

1. Start Application Explorer.

2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Configuring OracleAS Adapter for PeopleSoft" for information on creating a new configuration.

3. Select **Connect**.

   Nodes appear for Adapters, Events, and Business Services (also known as Web services).



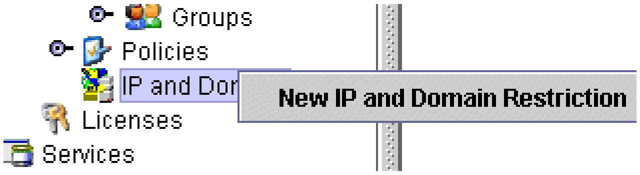
   Perform the following steps:

   a. Expand the **Business Services** node.

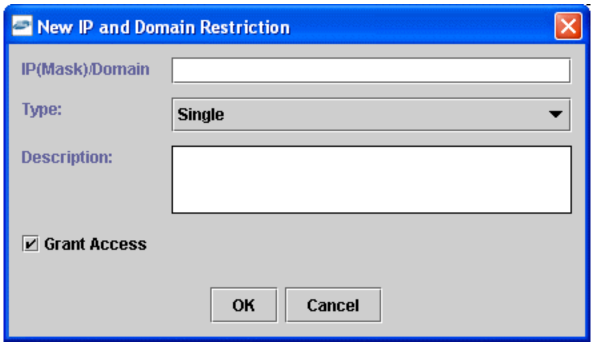
b. Expand the **Configuration** node.

c. Expand the **Security** node.

4. Right-click **IP and Domain** and select **New IP and Domain Restriction**.



The New IP and Domain Restriction dialog box is displayed.


New IP and Domain Restriction

IP(Mask)/Domain

Type: Single

Description:

☑ Grant Access

Perform the following steps:

a. In the **IP(Mask)/Domain** field, enter the IP or domain name using the following guidelines.

If you select **Single** (Computer) from the **Type** list, you must provide the IP address for that computer. If you only know the DNS name for the computer, click **DNS Lookup** to obtain the IP Address based on the DNS name.

If you select **Group** (of Computers), you must provide the IP address and subnet mask for the computer group.

If you select **Domain**, you must provide the domain name.

b. From the **Type** list, select the type of restriction.

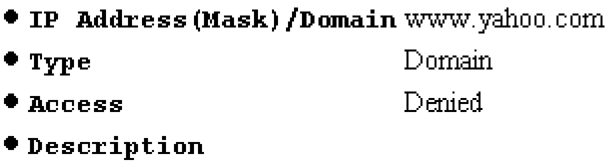
c. In the **Description** field, enter a description (optional).

d. To grant access, select the **Grant Access** check box.

5. Click **OK**.

The new domain is added under the IP and Domain node.

The following pane summarizes your configuration.


- **IP Address(Mask)/Domain** www.yahoo.com
- **Type** Domain
- **Access** Denied
- **Description**

Migrating Repositories

During design time, the Oracle repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at runtime. For management purposes, you can migrate BSE and J2CA repositories that are configured for Oracle to new destinations without affecting your existing configuration. For example, you may want to migrate a repository from a test environment to a production environment.

Migrating a BSE Repository

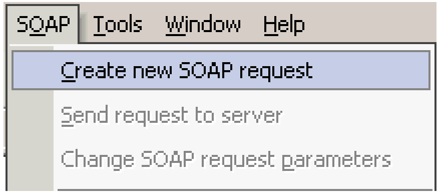
To migrate a BSE repository:

1. Copy the BSE control service URL, for example:

   ```
   http://localhost:7777/ibse/IBSEServlet/admin/iwcontrol.ibs
   ```

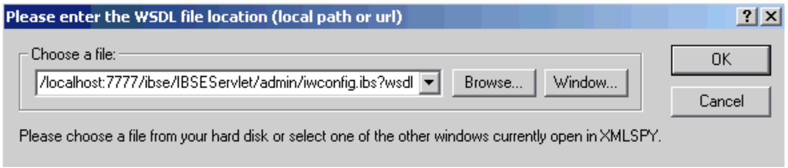
2. Open a third party XML editor, for example, XMLSPY.
3. From the menu bar, click **SOAP**.

   A list of options appears.



4. Select **Create new SOAP request**.

   The WSDL file location dialog box is displayed.



Choose a file: /localhost:7777/ibse/IBSEServlet/admin/iwconfig.ibs?wsdl

Please choose a file from your hard disk or select one of the other windows currently open in XMLSPY.

   Perform the following steps:

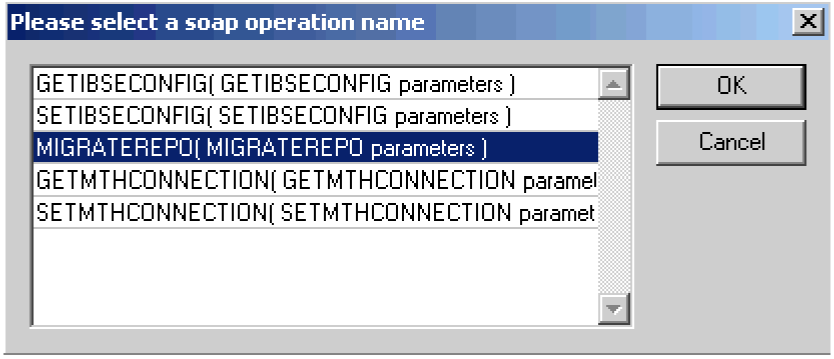
   a. In the **Choose a file** field, paste the BSE control service URL.

   b. Append **?wsdl** to the URL, for example:

   ```
   http://localhost:7777/ibse/IBSEServlet/admin/iwcontrol.ibs?wsdl
   ```

5. Click **OK**.

   The soap operation name dialog box is displayed, listing the available control methods.

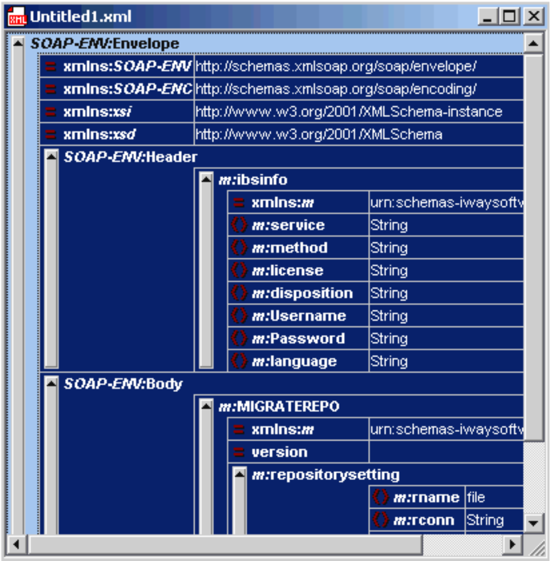


GETIBSECONFIG(GETIBSECONFIG parameters)
SETIBSECONFIG(SETIBSECONFIG parameters)
MIGRATEREPO(MIGRATEREPO parameters)
GETMTHCONNECTION(GETMTHCONNECTION paramet
SETMTHCONNECTION(SETMTHCONNECTION paramet

6. Select the **MIGRATEREPO(MIGRATEREPO parameters)** control method and click **OK**.

> **Note:** The **MIGRATEREPO(MIGRATEREPO parameters)** control method is available from the BSE administration console. This control method migrates all Web services to the new (empty) repository. You can choose to migrate select Web services only.

The following window is displayed. It shows the structure of the SOAP envelope.



SOAP-ENV:Envelope

xmlns:SOAP-ENV	http://schemas.xmlsoap.org/soap/envelope/
xmlns:SOAP-ENC	http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
xmlns:xsd	http://www.w3.org/2001/XMLSchema

SOAP-ENV:Header

m:ibsinfo

xmlns:m	urn:schemas-iwaysoftv
m:service	String
m:method	String
m:license	String
m:disposition	String
m:Username	String
m:Password	String
m:language	String

SOAP-ENV:Body

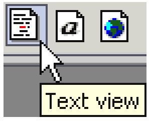
m:MIGRATEREPO

xmlns:m	urn:schemas-iwaysoftv
version	

m:repositorysetting

m:rname	file
m:rconn	String

7. Locate the **Text view** icon in the toolbar.



8. To display the structure of the SOAP envelope as text, click the **Text view** icon.

   The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

9. Locate the following section:

```
<m:MIGRATEREPO xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config"
version="">
```

```
<m:repositorysetting>
<m:rname>oracle</m:rname>
<m:rconn>String</m:rconn>
<m:rdriver>String</m:rdriver>
<m:ruser>String</m:ruser>
<m:rpwd>String</m:rpwd>
</m:repositorysetting>
<m:servicename>String</m:servicename>
</m:MIGRATEREPO>
```

Perform the following steps:

a. For the **<m:rconn>** tag, replace the String placeholder with a repository URL where you want to migrate your existing BSE repository.

   The Oracle repository URL has the following format:

   ```
   jdbc:oracle:thin:@[host]:[port]:[sid]
   ```

b. For the **<m:rdriver>** tag, replace the String placeholder with the location of your Oracle driver.

c. For the **<m:ruser>** tag, replace the String placeholder with a valid user name to access the Oracle repository.

d. For the **<m:rpwd>** tag, replace the String placeholder with a valid password to access the Oracle repository.

10. Perform one of the following migration options.

    - If you want to migrate a single Web service from the current BSE repository, enter the Web service name in the `<m:servicename>` tag, for example:

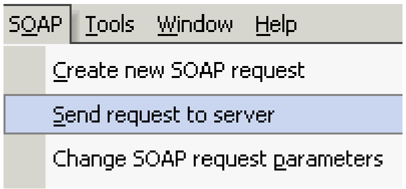
      ```
      <m:servicename>PeopleSoftService1</m:servicename>
      ```

    - If you want to migrate multiple Web services from the current BSE repository, duplicate the `<m:servicename>` tag for each Web service, for example:

      ```
      <m:servicename>PeopleSoftService1</m:servicename>
      <m:servicename>PeopleSoftService2</m:servicename>
      ```

    - If you want to migrate all Web services from the current BSE repository, remove the `<m:servicename>` tag.

11. From the menu bar, click **SOAP** and select **Send request to server**, as shown in the following image.



Your BSE repository and any Web services you specified are now migrated to the new Oracle repository URL you specified.

Migrating a J2CA Repository

To migrate a J2CA repository:

1. Navigate to the location of your J2CA configuration directory where the repository schemas and other information is stored, for example:

```
OracleAS_home\adapters\application\config\JCA_CONFIG
```

Where `JCA_CONFIG` is the name of your J2CA configuration.

2. Locate and copy the `repository.xml` file.
3. Place this file in a new J2CA configuration directory to migrate the existing repository.

   Your J2CA repository is migrated to the new J2CA configuration directory.

A

Generating Component Interface APIs

This section describes how to generate component interface APIs to use with Oracle Application Server Adapter for PeopleSoft.

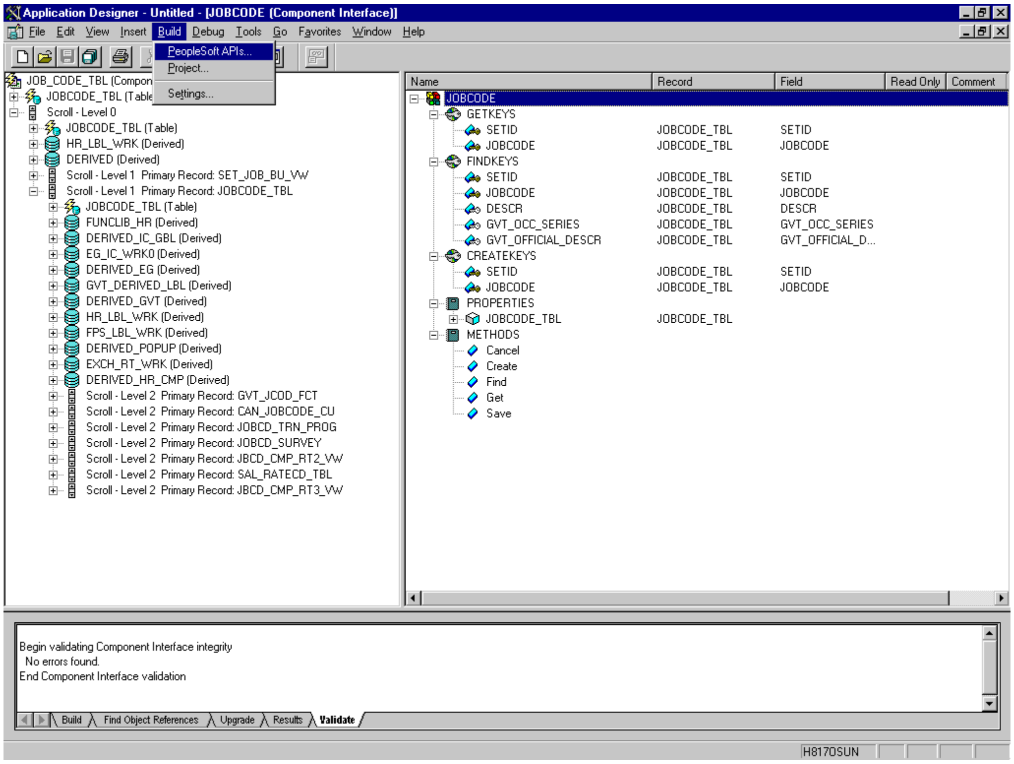
Building the PeopleSoft API Java Programs

Whether you are using an Enterprise Integration Point (EIP) supplied by PeopleSoft or a customized component interface, you must create a PeopleSoft API to enable communications with the PeopleSoft application. The API is a collection of Java class files that reside on the client machine and mediate between the client application layer and PeopleSoft.

Before using your component interface, you must apply security to it and test it.

To build a PeopleSoft API Java program:

1. Open the PeopleSoft Application Designer.



```
JOB_CODE_TBL (Compon
JOBCODE_TBL (Table
Scroll - Level 0
    JOBCODE_TBL (Table)
    HR_LBL_WRK (Derived)
    DERIVED (Derived)
    Scroll - Level 1  Primary Record: SET_JOB_BU_VW
    Scroll - Level 1  Primary Record: JOBCODE_TBL
        JOBCODE_TBL (Table)
        FUNCLIB_HR (Derived)
        DERIVED_IC_GBL (Derived)
        EG_IC_WRK0 (Derived)
        DERIVED_EG (Derived)
        GVT_DERIVED_LBL (Derived)
        DERIVED_GVT (Derived)
        HR_LBL_WRK (Derived)
        FPS_LBL_WRK (Derived)
        DERIVED_POPUP (Derived)
        EXCH_RT_WRK (Derived)
        DERIVED_HR_CMP (Derived)
        Scroll - Level 2  Primary Record: GVT_JCOD_FCT
        Scroll - Level 2  Primary Record: CAN_JOBCODE_CU
        Scroll - Level 2  Primary Record: JOBCD_TRN_PROG
        Scroll - Level 2  Primary Record: JOBCD_SURVEY
        Scroll - Level 2  Primary Record: JBCD_CMP_RT2_VW
        Scroll - Level 2  Primary Record: SAL_RATECD_TBL
        Scroll - Level 2  Primary Record: JBCD_CMP_RT3_VW
```

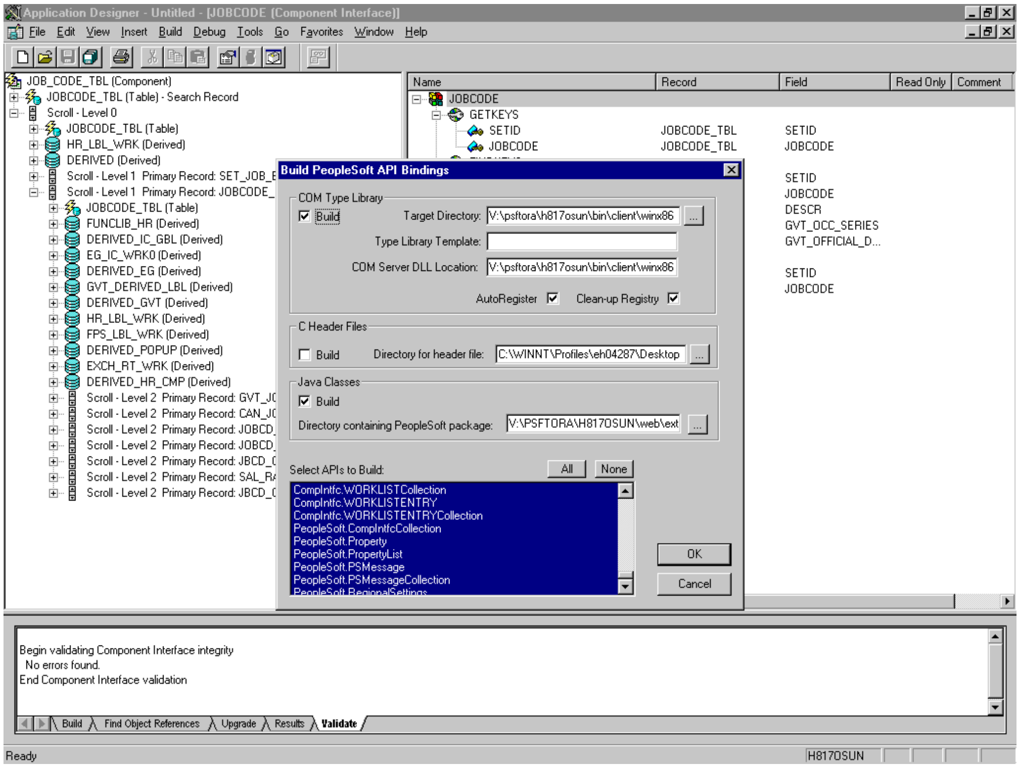
Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
DESCR	JOBCODE_TBL	DESCR		
GVT_OCC_SERIES	JOBCODE_TBL	GVT_OCC_SERIES		
GVT_OFFICIAL_DESCR	JOBCODE_TBL	GVT_OFFICIAL_D...		
CREATEKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
PROPERTIES				
JOBCODE_TBL	JOBCODE_TBL			
METHODS				
Cancel				
Create				
Find				
Get				
Save				

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

2. From the PeopleSoft Application Designer, open a component interface.

3. Click the right pane and select **PeopleSoft APIs** from the **Build** menu.

   The Build PeopleSoft API Bindings dialog box prompts you for the types of bindings to create.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_E
Scroll - Level 1 Primary Record: JOBCODE_
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JC
Scroll - Level 2 Primary Record: CAN_JC
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JBCD_C
Scroll - Level 2 Primary Record: SAL_RA
Scroll - Level 2 Primary Record: JBCD_C
Name
Record
Field
Read Only
Comment
JOBCODE
GETKEYS
SETID JOBCODE_TBL SETID
JOBCODE JOBCODE_TBL JOBCODE
SETID
JOBCODE
DESCR
GVT_OCC_SERIES
GVT_OFFICIAL_D...
SETID
JOBCODE
Build Find Object References Upgrade Results Validate
Ready
H817OSUN

Build PeopleSoft API Bindings

COM Type Library

☑ Build Target Directory: V:\psftora\h817osun\bin\client\winx86 ...

Type Library Template:

COM Server DLL Location: V:\psftora\h817osun\bin\client\winx86

AutoRegister ☑ Clean-up Registry ☑

C Header Files

☐ Build Directory for header file: C:\WINNT\Profiles\eh04287\Desktop ...

Java Classes

☑ Build

Directory containing PeopleSoft package: V:\PSFTORA\H817OSUN\web\ext ...

Select APIs to Build: All None

CompIntfc.WORKLISTCollection
CompIntfc.WORKLISTENTRY
CompIntfc.WORKLISTENTRYCollection
PeopleSoft.CompIntfcCollection
PeopleSoft.Property
PeopleSoft.PropertyList
PeopleSoft.PSMessage
PeopleSoft.PSMessageCollection
PeopleSoft.RegionalSettings

OK Cancel

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

4. Because you are creating Java files, ensure you deselect **COM Type Library Build**.

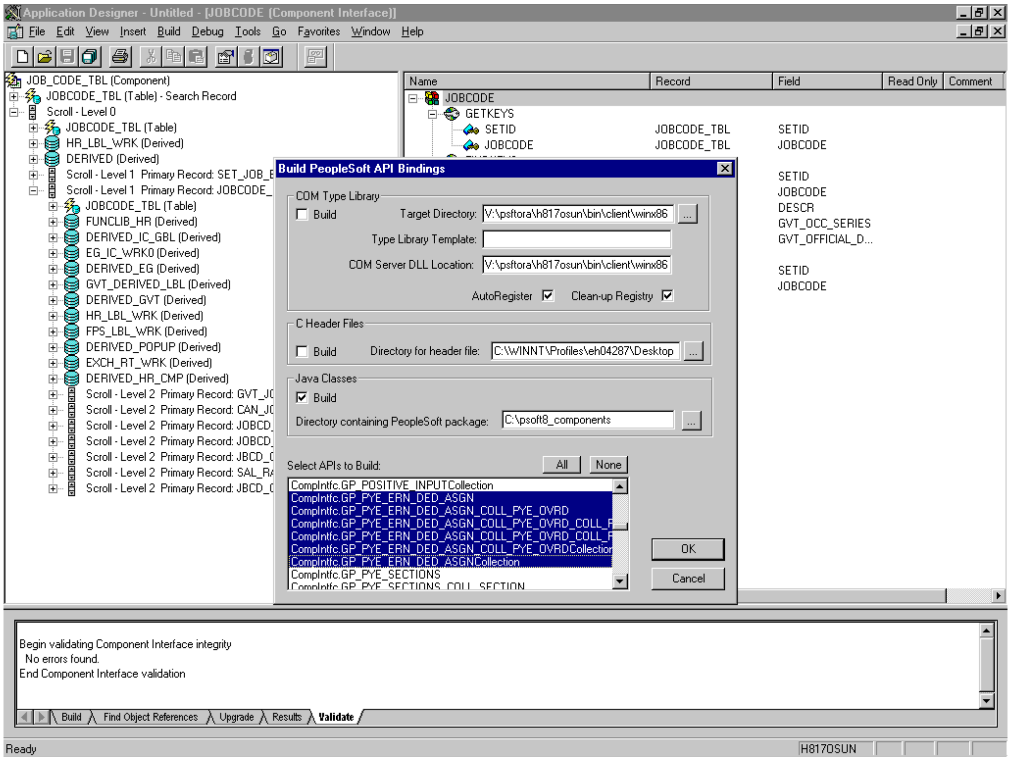
5. Ensure that **Java Classes Build** is selected, and then select a directory on your local machine where the Java files are to be placed, for example, `c:\psoft8_components`.

   - To build all API files, select the default, **All** (potentially a large number), and click **OK**.

     PeopleSoft generates the files. This takes a few minutes. After the process is complete, a message appears in the output window.

     You are now ready to compile the Java files. See "Compiling the PeopleSoft API Java Programs" on page A-4 for more information.

     The following pane illustrates the `GP_PYE_ERN_DED_ASGN` component interface from the HR 8.1 application.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_E
Scroll - Level 1 Primary Record: JOBCODE_
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JC
Scroll - Level 2 Primary Record: CAN_JC
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JBCD_C
Scroll - Level 2 Primary Record: SAL_RA
Scroll - Level 2 Primary Record: JBCD_C
Name
Record
Field
Read Only
Comment
JOBCODE
GETKEYS
SETID
JOBCODE
SETID
JOBCODE
DESCR
GVT_OCC_SERIES
GVT_OFFICIAL_D...
SETID
JOBCODE
All
None
OK
Cancel
Build
Find Object References
Upgrade
Results
Validate
Ready
H817OSUN

Name	Record	Field
SETID	JOBCODE_TBL	SETID
JOBCODE	JOBCODE_TBL	JOBCODE

Build PeopleSoft API Bindings

COM Type Library

☐ Build Target Directory: V:\psftora\h817osun\bin\client\winx86

Type Library Template:

COM Server DLL Location: V:\psftora\h817osun\bin\client\winx86

AutoRegister ☑ Clean-up Registry ☑

C Header Files

☐ Build Directory for header file: C:\WINNT\Profiles\eh04287\Desktop

Java Classes

☑ Build

Directory containing PeopleSoft package: C:\psoft8_components

Select APIs to Build:

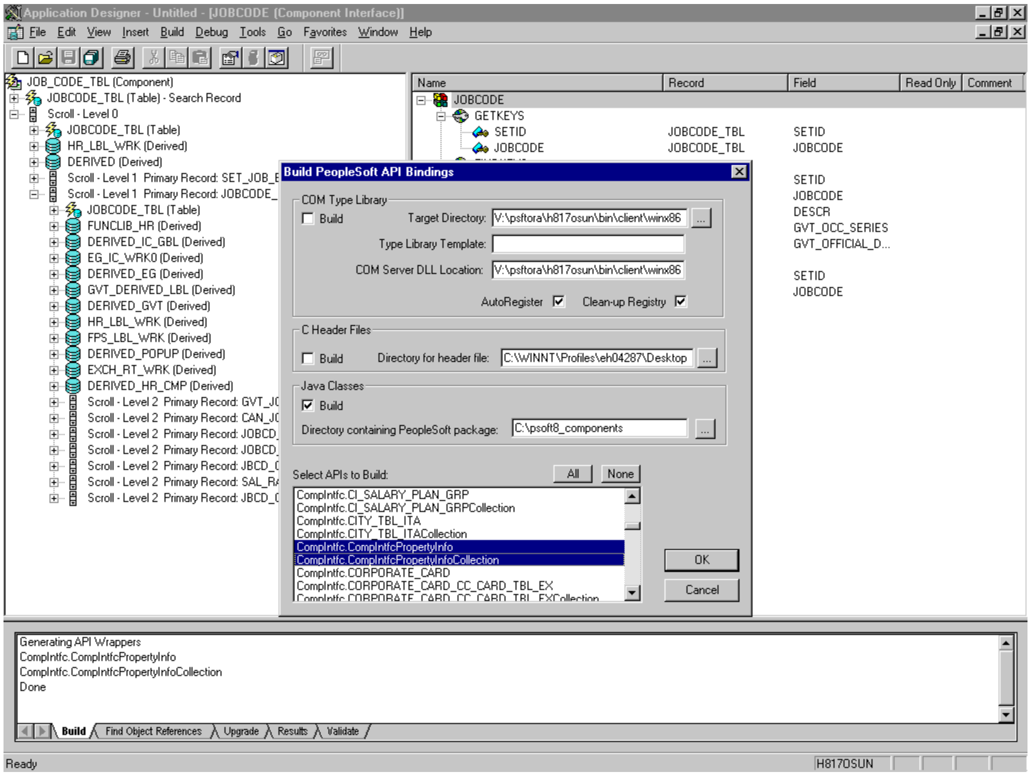
CompIntfc.GP_POSITIVE_INPUTCollection
CompIntfc.GP_PYE_ERN_DED_ASGN
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRD
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRD_COLL_F
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRD_COLL_F
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRDCollection
CompIntfc.GP_PYE_ERN_DED_ASGNCollection
CompIntfc.GP_PYE_SECTIONS
CompIntfc.GP_PYE_SECTIONS_COLL_SECTION

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

- To create APIs for a specific component interface or interfaces, click **None**.

  This clears the selected APIs, so you can select the appropriate APIs for your component interface. These APIs begin with the name of your component interface. There may be fewer than five, or more than 50 APIs, for a particular component interface.

  The following dialog box displays the following APIs, including generic component interface properties.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
Name
Record
Field
Read Only
Comment
JOBCODE
GETKEYS
SETID
JOBCODE
Ready
H817OSUN

Build PeopleSoft API Bindings

COM Type Library
☐ Build Target Directory: V:\psftora\h817osun\bin\client\winx86
Type Library Template:
COM Server DLL Location: V:\psftora\h817osun\bin\client\winx86
AutoRegister ☑ Clean-up Registry ☑

C Header Files
☐ Build Directory for header file: C:\WINNT\Profiles\eh04287\Desktop

Java Classes
☑ Build
Directory containing PeopleSoft package: C:\psoft8_components

Select APIs to Build: All None
CompIntfc.CI_SALARY_PLAN_GRP
CompIntfc.CI_SALARY_PLAN_GRPCollection
CompIntfc.CITY_TBL_ITA
CompIntfc.CITY_TBL_ITACollection
CompIntfc.CompIntfcPropertyInfo
CompIntfc.CompIntfcPropertyInfoCollection
CompIntfc.CORPORATE_CARD
CompIntfc.CORPORATE_CARD_CC_CARD_TBL_EX
CompIntfc.CORPORATE_CARD_CC_CARD_TBL_EXCollection

OK Cancel

```
Generating API Wrappers
CompIntfc.CompIntfcPropertyInfo
CompIntfc.CompIntfcPropertyInfoCollection
Done
```

Build / Find Object References / Upgrade / Results / Validate

a. In addition to the APIs for the selected component interface, you also must generate the API files for the following generic component interface properties:

`CompIntfcPropertyInfo`

`CompIntfcPropertyInfoCollection`

You may select these items in the same step as the component interface build, or you may select them separately.

b. Click **OK**.

PeopleSoft generates the files. This takes a few minutes. After the process is complete, a message appears in the output window. You are now ready to compile the Java files. See "Compiling the PeopleSoft API Java Programs" on page A-4 for more information.

Compiling the PeopleSoft API Java Programs

PeopleSoft places the Java programs to compile in the directory called *`psoft8_components`*`\PeopleSoft\Generated\CompIntfc`.

Where `psoft8_components` is the directory specified during the build process.

If you chose to generate all APIs, the systems creates a second directory, *`psoft8_components`*`\PeopleSoft\Generated\PeopleSoft`. You are not required to access it.

The process for compiling the PeopleSoft API Java programs depends on whether you are compiling on the machine where you installed Application Explorer or on another machine.

> **Note:** There are two Java programs for every API file that you selected when you built the Java programs. See "Building the PeopleSoft API Java Programs" on page A-1 for more information.

Before you compile the Java programs, you require the PeopleSoft Java Object Adapter, the `psjoa.jar` file that resides on your PeopleSoft Application Server under the *`PS_HOME`*`\Web\psjoa` directory. This is the file that you placed in the adapter lib directory during installation.

If you are compiling on the same machine where you installed Application Explorer

Point to the `psjoa.jar` file or copy it to the directory where you placed the Java API files, for example, `c:\psoft8_components`.

If you are compiling on a machine other than the one where you installed Application Explorer

Perform the following steps:

1. Obtain a copy of the `psjoa.jar` file from the PeopleSoft Application Server. Ensure that the `psjoa.jar` file is in the Java class path before you compile the programs.
2. Compile the Java programs and ensure that you include the \PeopleSoft\Generated\CompIntfc path.

   The path is case-sensitive.

   The following Windows batch file, run from the `psoft8_components` directory, properly compiles the Java APIs. The code assumes that `psjoa.jar` was placed in `psoft8_components`.

   ```
   @echo off
   set JAVA_HOME=<my-java-home>
   set PATH=%JAVA_HOME%\bin;%PATH%
   set CLASSPATH=%JAVA_HOME%\lib\tools.jar;psjoa.jar;%CLASSPATH%
   javac -classpath %CLASSPATH% .\PeopleSoft\Generated\CompIntfc\*.java
   ```

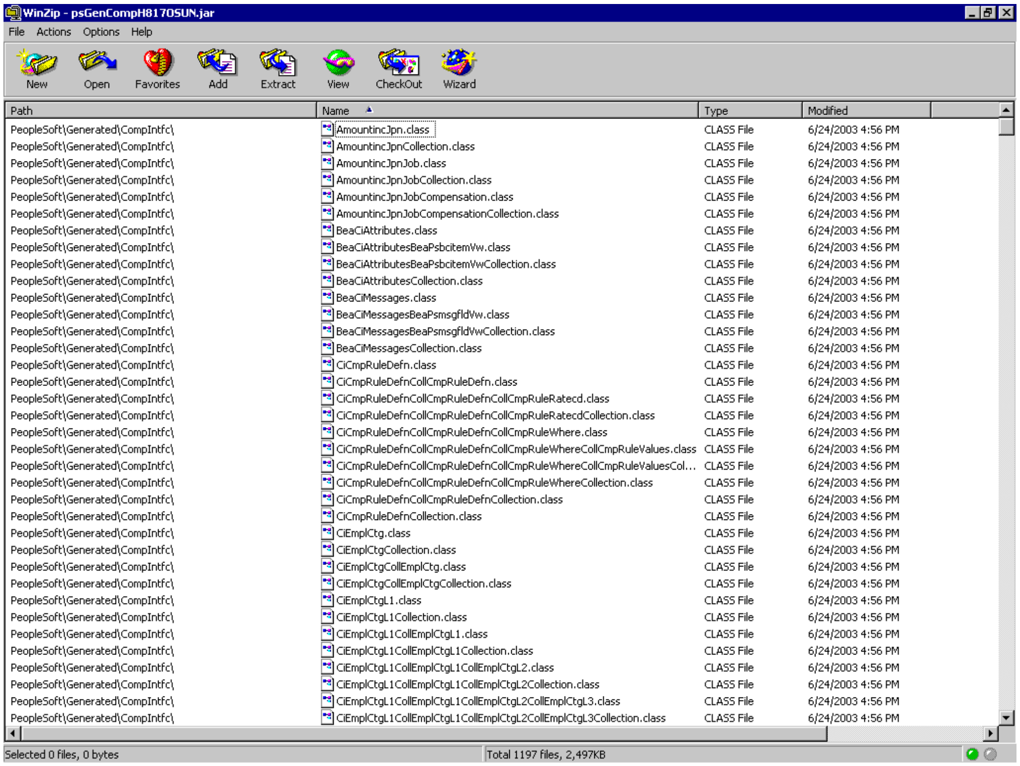
   Where `<my-java-home>` is the fully qualified path name of your Java home directory.

   This code places the class files in the same directory with the Java files, but you can choose a different location depending on your site requirements.
3. Compress the class files into a JAR file.

   The following Windows batch file, if run from the `psoft8_components` directory, creates a correct JAR file:

   ```
   @echo off
   set JAVA_HOME= my-java-home
   set PATH=%JAVA_HOME%\bin;%PATH%
   set CLASSPATH=%JAVA_HOME%\lib\tools.jar;%CLASSPATH%
   jar cvf my-jar-file.jar .\PeopleSoft\Generated\CompIntfc\*.class
   ```

   Where appropriate, substitutions are made for `my-java-home` and `my-jar-file`.
4. To verify that your JAR file is correct, open it with the WinZip application.



Path	Name	Type	Modified
PeopleSoft\Generated\CompIntfc\	AmountincJpn.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJob.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJobCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJobCompensation.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJobCompensationCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributes.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributesBeaPsbcitemVw.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributesBeaPsbcitemVwCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributesCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessages.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessagesBeaPsmsgfldVw.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessagesBeaPsmsgfldVwCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessagesCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefn.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefn.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleRatecd.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleRatecdCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhere.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhereCollCmpRuleValues.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhereCollCmpRuleValuesCol...	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhereCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtg.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgCollEmplCtg.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgCollEmplCtgCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1Collection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1Collection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2Collection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2CollEmplCtgL3.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2CollEmplCtgL3Collection.class	CLASS File	6/24/2003 4:56 PM

Selected 0 files, 0 bytes

Total 1197 files, 2,497KB

If the JAR file does not use the case-sensitive `PeopleSoft\Generated\CompIntfc\` path, you must go back and correct it.

5. Place the JAR file in the adapters common lib directory. This enables the OracleAS Adapter for PeopleSoft to communicate with the PeopleSoft component interface.

   OracleAS_home`\adapters\application\lib`

Note: If you are running on UNIX, do the compile and JAR steps on Windows and then move the file to your UNIX machine. The JAR file is binary. If you use an FTP-based tool to move your JAR file from Windows to UNIX, the file format must be set to binary.

B

Configuring the PeopleSoft Message Router

This section describes how to configure and test a TCP/IP or HTTP target connector and a TCP/IP handler for PeopleSoft.

The following configuration topics assume you are familiar with PeopleSoft Integration Broker (in release 8.4) or Application Messaging (in release 8.1). If not, see Appendix D, "Using PeopleSoft Integration Broker" for more information. For a complete description *before* you work with OracleAS Adapter for PeopleSoft, see your PeopleSoft documentation.

> **Note:** In PeopleSoft release 8.1, the messaging architecture is called Application Messaging and includes Application Messaging Gateway. In release 8.4, the messaging architecture is called Integration Broker, which includes Integration Gateway. When discussing release-independent issues, this section uses release 8.4 terminology. When discussing release-specific issues, it uses release-specific terminology.

Configuring the TCP/IP or HTTP Target Connector for PeopleSoft 8.4

The procedures in this topic assume that your Integration Broker environment is configured and tested. See Appendix D, "Using PeopleSoft Integration Broker" for more information.

1. Configure the gateway for the TCP/IP Target Connector or HTTP Target Connector. See Configuring the TCP/IP Target Connector on page B-2 for more information.

   > **Note:** This step is optional when configuring the HTTP Connector. The HTTP Target Connector is supplied with your PeopleSoft application, and no special configuration steps are required. If you choose, you may configure default connection values on the Gateway. You can override these values when you configure the node.

2. Configure the node. See Configuring the Node for the TCP/IP84 Connector on page B-3 for more information.

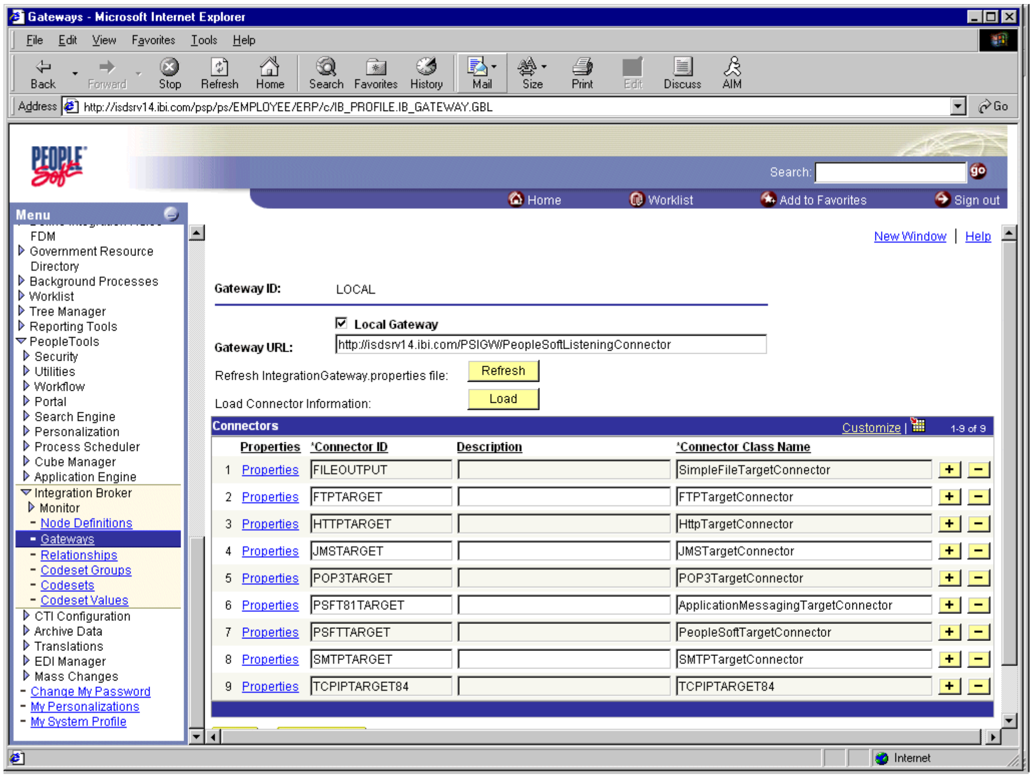
   > **Note:** Starting with release 8.4, the Integration Broker is delivered with an HTTP Outbound Connector. See Configuring the HTTP Connector on page B-7 for more information.

Configuring the TCP/IP Target Connector

To configure the gateway for the TCP/IP Target Connector:

1. In a Web browser, open your PeopleSoft release 8.4 application.
2. In the menu pane, expand **PeopleTools**, then expand **Integration Broker**, and click **Gateways**.
3. Open the **LOCAL Gateway ID**.

   A pane similar to the following Gateway ID pane is displayed.



Gateway ID: LOCAL

☑ Local Gateway

Gateway URL: http://isdsrv14.ibi.com/PSIGW/PeopleSoftListeningConnector

Refresh IntegrationGateway.properties file: Refresh

Load Connector Information: Load

Connectors — Customize | 1-9 of 9

	Properties	*Connector ID	Description	*Connector Class Name
1	Properties	FILEOUTPUT		SimpleFileTargetConnector
2	Properties	FTPTARGET		FTPTargetConnector
3	Properties	HTTPTARGET		HttpTargetConnector
4	Properties	JMSTARGET		JMSTargetConnector
5	Properties	POP3TARGET		POP3TargetConnector
6	Properties	PSFT81TARGET		ApplicationMessagingTargetConnector
7	Properties	PSFTTARGET		PeopleSoftTargetConnector
8	Properties	SMTPTARGET		SMTPTargetConnector
9	Properties	TCPIPTARGET84		TCPIPTARGET84

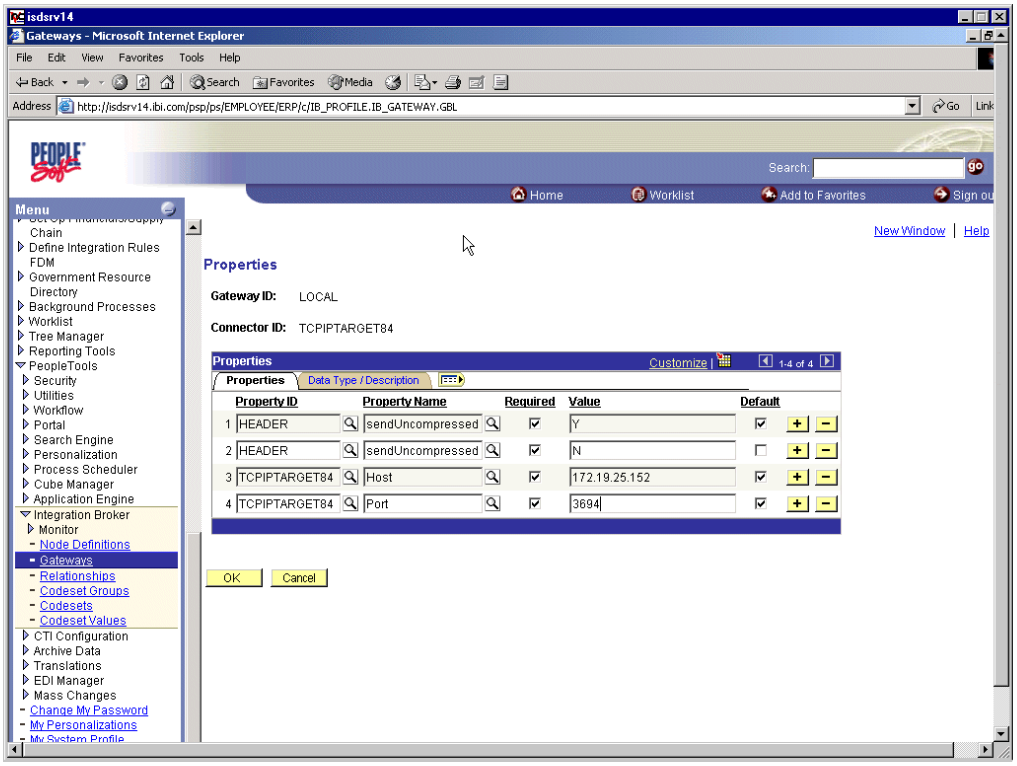
4. If you do not see the TCPIPTARGET84 Connector ID, click **Load** and scroll to locate **TCPIPTARGET84** in the list.

   If TCPIPTARGET84 still does not appear, the connector class file was not installed in the Integration Gateway.

   Perform the following steps:

   a. Click the **Properties** URL for TCPIPTARGET84.

      The Properties pane for TCPIPTARGET84 is displayed. Default values appear for the host and the port. For complex business situations, you can override this setting on the individual node.



Properties

Gateway ID: LOCAL

Connector ID: TCPIPTARGET84

Properties

	Property ID	Property Name	Required	Value	Default
1	HEADER	sendUncompressed	☑	Y	☑
2	HEADER	sendUncompressed	☑	N	☐
3	TCPIPTARGET84	Host	☑	172.19.25.152	☑
4	TCPIPTARGET84	Port	☑	3694	☑

b. Enter the values for the host and the port for the machine on which your PeopleSoft XML listener is listening for incoming messages.

5. Click **OK**.

   The Gateway window is displayed.

6. Scroll to the bottom of the window and click **Save**.

You have finished configuring the gateway for the TCP/IP Target Connector.

Configuring the Node for the TCP/IP84 Connector

To configure the node for the TCP/IP84 Connector:

1. In the Menu pane, select **PeopleTools**, **Integration Broker**, and then click **Node Definitions**.

2. Select the node that you want to configure.

   Note: This procedure uses a node called EXTERNAL. For more information about creating and using nodes, see Appendix D, "Using PeopleSoft Integration Broker" or your PeopleSoft documentation.

   Perform the following steps:

   a. Select **External** from the **Node Type** list.

   b. Select **Implicit** from the **Routing Type** list.

3. Select the **Connectors** tab.

   Perform the following steps:

   a. Select **TCPIPTARGET84** as the Connector ID.

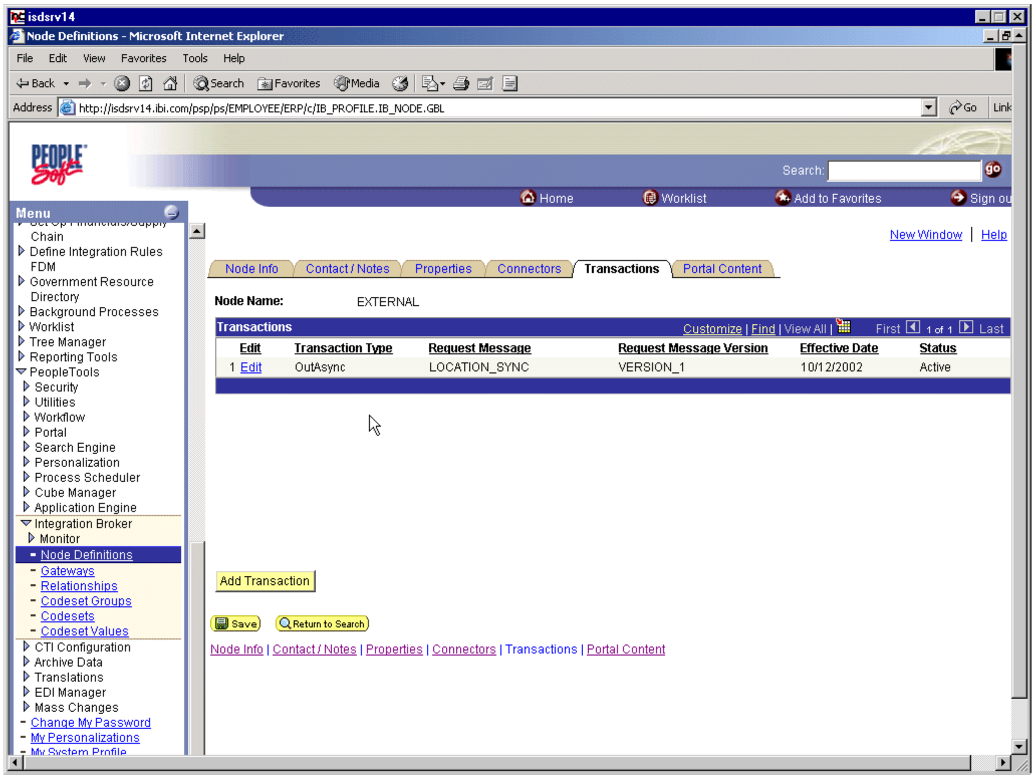
Default values appear for the host and the port.

b. Enter the values for the host and the port for the machine and port that route XML to Oracle Application Server. You can accept or override the default values for individual nodes.

c. Click **Save**.

4. If you are warned that you are changing the connector, click **OK**.

5. Select the **Transactions** tab.



Node Name: EXTERNAL

Transactions

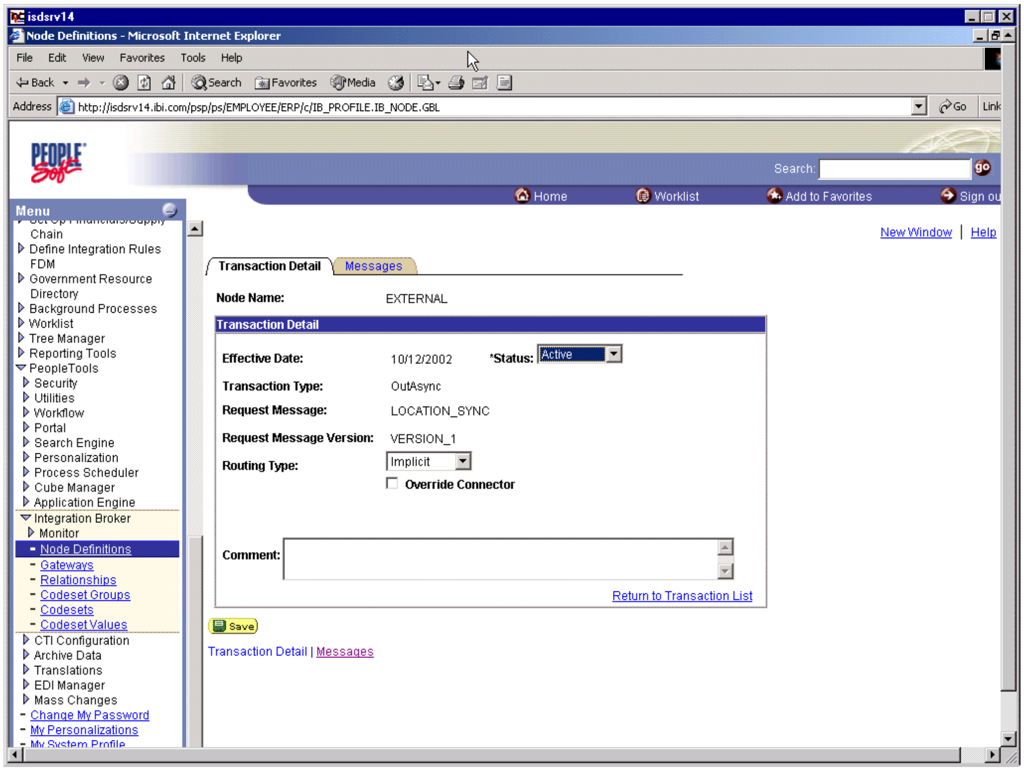
	Edit	Transaction Type	Request Message	Request Message Version	Effective Date	Status
1	Edit	OutAsync	LOCATION_SYNC	VERSION_1	10/12/2002	Active

Perform the following steps:

a. If there are no transactions, click **Add Transaction** to add the message with which you are working. In this procedure, the node is already configured with the `LOCATION_SYNC` message.

b. To view transaction details for the `LOCATION_SYNC` message, click **Edit**.

The Transaction Detail tab appears



Node Name: EXTERNAL

Transaction Detail

Effective Date: 10/12/2002 *Status: Active

Transaction Type: OutAsync

Request Message: LOCATION_SYNC

Request Message Version: VERSION_1

Routing Type: Implicit

□ Override Connector

Comment:

Return to Transaction List

c. Add the message with which you are working.

d. Verify that the **Routing Type** is **Implicit**.

6. Click **Save**.

7. Return to the **Transactions** tab.

   Perform the following steps:

   a. Edit additional transactions by clicking **Edit** and navigating to the **Transaction Detail** tab.

   b. In the **Transaction Detail** tab, select **Inactive** from the **Status** list.

      Inactive status is for initial testing only. After you test your configuration, you may change the status to Active and have as many nodes and transactions as required to satisfy your business requirements.

8. Click **Save**.

You can now send XML messages to your PeopleSoft XML listener.

Configuring the HTTP Target Connector

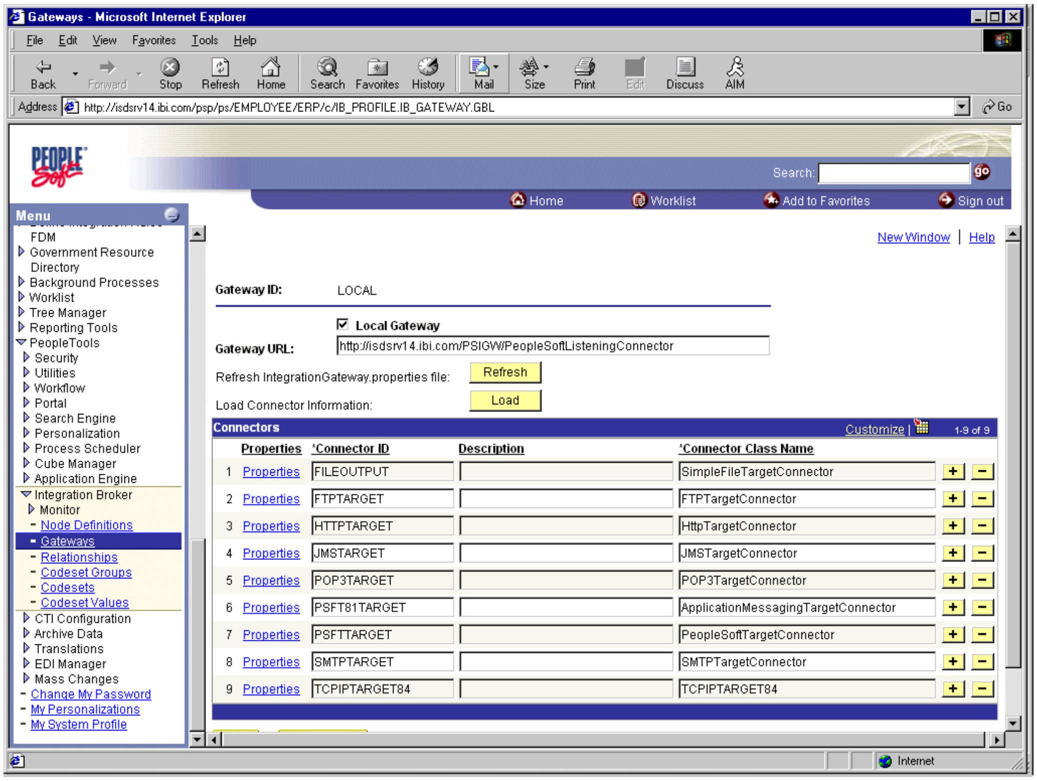
The HTTP Target Connector is supplied with your PeopleSoft application, and no special configuration steps are required. If you choose, you may configure default connection values on the Gateway. You can override these values when you configure the node. To configure the gateway for the HTTP Target Connector:

1. In a Web browser, open your PeopleSoft 8.4 application.

2. In the Menu pane, expand **PeopleTools**, then expand **Integration Broker**, and click **Gateways**.

3. Open the **LOCAL Gateway ID**.

A pane similar to the following Gateway ID pane is displayed.



Gateway ID: LOCAL

☑ Local Gateway

Gateway URL: http://isdsrv14.ibi.com/PSIGW/PeopleSoftListeningConnector

Refresh IntegrationGateway.properties file: Refresh

Load Connector Information: Load

Connectors — Customize | 1-9 of 9

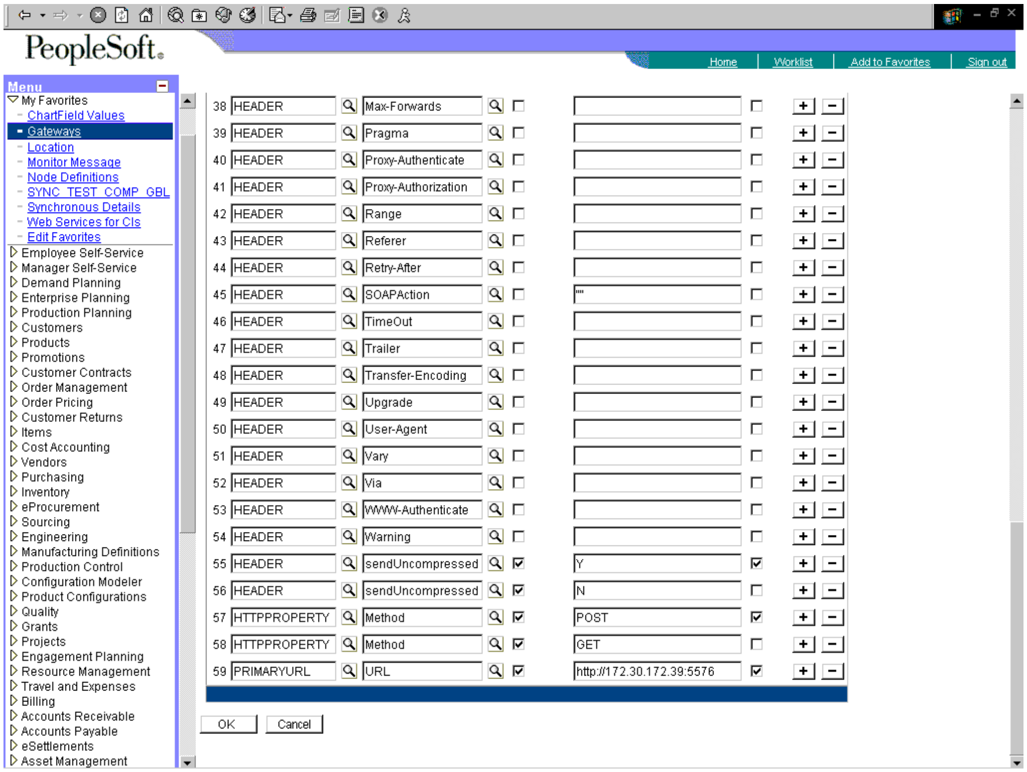
	Properties	*Connector ID	Description	*Connector Class Name		
1	Properties	FILEOUTPUT		SimpleFileTargetConnector	+	−
2	Properties	FTPTARGET		FTPTargetConnector	+	−
3	Properties	HTTPTARGET		HttpTargetConnector	+	−
4	Properties	JMSTARGET		JMSTargetConnector	+	−
5	Properties	POP3TARGET		POP3TargetConnector	+	−
6	Properties	PSFT81TARGET		ApplicationMessagingTargetConnector	+	−
7	Properties	PSFTTARGET		PeopleSoftTargetConnector	+	−
8	Properties	SMTPTARGET		SMTPTargetConnector	+	−
9	Properties	TCPIPTARGET84		TCPIPTARGET84	+	−

4. If you do not see the **HTTPTARGET Connector ID**, click **Load**.

   If it does not appear, your Gateway was not installed properly. Check with your PeopleSoft system administrator.

5. Click the **Properties** URL for HTTPTARGET.

   The Properties pane for HTTPTARGET displays default values:



38	HEADER	Max-Forwards	□		□	+ −
39	HEADER	Pragma	□		□	+ −
40	HEADER	Proxy-Authenticate	□		□	+ −
41	HEADER	Proxy-Authorization	□		□	+ −
42	HEADER	Range	□		□	+ −
43	HEADER	Referer	□		□	+ −
44	HEADER	Retry-After	□		□	+ −
45	HEADER	SOAPAction	□	""	□	+ −
46	HEADER	TimeOut	□		□	+ −
47	HEADER	Trailer	□		□	+ −
48	HEADER	Transfer-Encoding	□		□	+ −
49	HEADER	Upgrade	□		□	+ −
50	HEADER	User-Agent	□		□	+ −
51	HEADER	Vary	□		□	+ −
52	HEADER	Via	□		□	+ −
53	HEADER	WWW-Authenticate	□		□	+ −
54	HEADER	Warning	□		□	+ −
55	HEADER	sendUncompressed	☑	Y	☑	+ −
56	HEADER	sendUncompressed	☑	N	□	+ −
57	HTTPPROPERTY	Method	☑	POST	☑	+ −
58	HTTPPROPERTY	Method	☑	GET	□	+ −
59	PRIMARYURL	URL	☑	http://172.30.172.39:5576	☑	+ −

6. Scroll to the bottom and enter a value for the **PRIMARYURL**.

   This is the default HTTP address (machine and port) on which your PeopleSoft XML listener is listening for incoming messages.

   > **Note:** For complex business situations, you can override this setting on the individual node.

7. Click **OK**.

   The Gateway window is displayed.

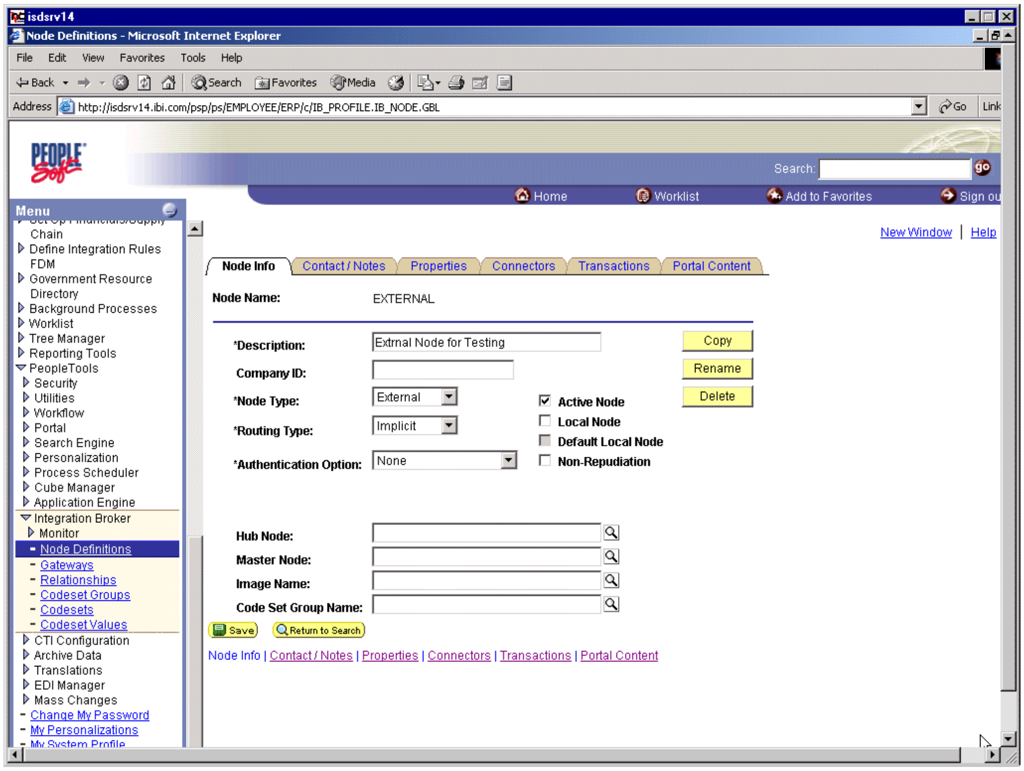
8. Scroll to the bottom of the window and click **Save**.

You have finished configuring the gateway for the HTTP Target Connector.

Configuring the HTTP Connector

Starting with release 8.4, the Integration Broker is delivered with an HTTP Outbound Connector. This connector can be used in place of the TCP/IP84 connector for sending messages to Oracle Application Server.

To configure the node to use the HTTP Connector:



Address: http://isdsrv14.ibi.com/psp/ps/EMPLOYEE/ERP/c/IB_PROFILE.IB_NODE.GBL

Node Name:	EXTERNAL
*Description:	Extrnal Node for Testing
Company ID:	
*Node Type:	External
*Routing Type:	Implicit
*Authentication Option:	None
Hub Node:	
Master Node:	
Image Name:	
Code Set Group Name:	

☑ Active Node
☐ Local Node
☐ Default Local Node
☐ Non-Repudiation

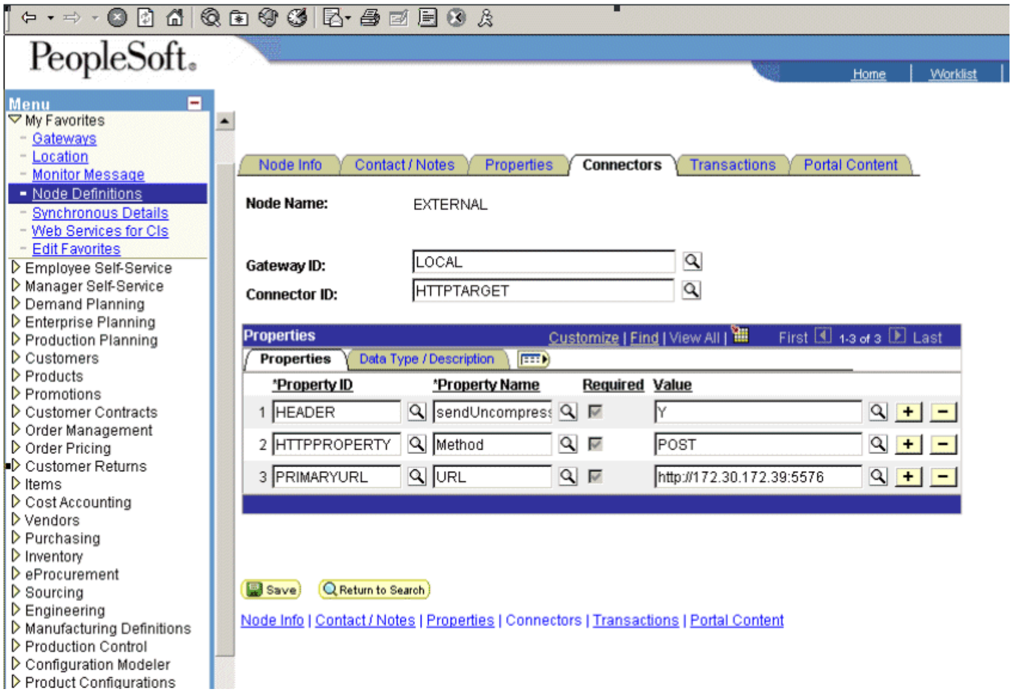
1. In the Menu pane, expand **PeopleTools**, **Integration Broker**, and then click **Node Definitions**.
2. Select the node that you want to configure.

> **Note:** This procedure uses a node called EXTERNAL. For more information about creating and using nodes, see Appendix D, "Using PeopleSoft Integration Broker" or your PeopleSoft documentation.

Perform the following steps:

a. From the **Node Type** list, select **External**.

b. From the **Routing Type** list, select **Implicit**.



Node Name: EXTERNAL

Gateway ID: LOCAL

Connector ID: HTTPTARGET

Properties

	*Property ID	*Property Name	Required	Value
1	HEADER	sendUncompress	☑	Y
2	HTTPPROPERTY	Method	☑	POST
3	PRIMARYURL	URL	☑	http://172.30.172.39:5576

3. Select the **Connectors** tab.

   Perform the following steps:

   a. Change the Connector ID to **HTTPTARGET**.

   b. Enter a value for each property based on the following:

Property ID	Property Name	Value
HEADER	sendUncompressed	Y
HTTPPROPERTY	Method	POST
PRIMARYURL	URL	URL and the port of the HTTP listener

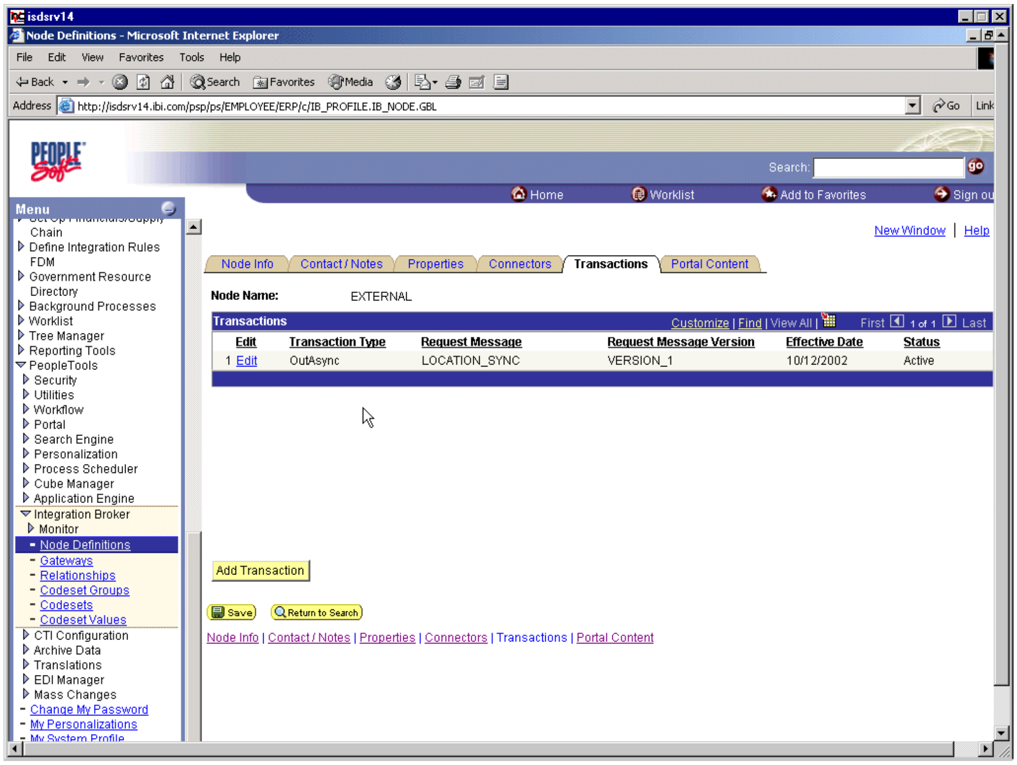
> **Note:** For complex business situations you can configure multiple nodes and multiple listeners.

4. Click **Save**.

5. If you are warned that you are changing the Connector, click **OK**.

6. Select the **Transactions** tab.

   The following pane is displayed.



Node Name: EXTERNAL

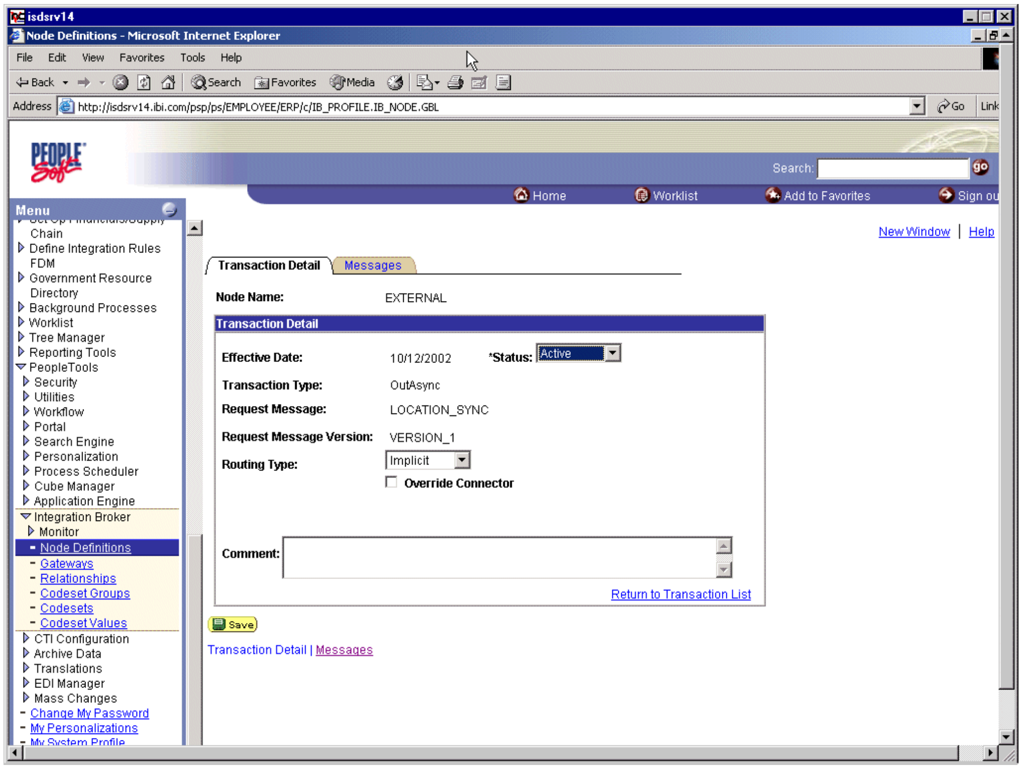
Transactions — Customize | Find | View All | First 1 of 1 Last

	Edit	Transaction Type	Request Message	Request Message Version	Effective Date	Status
1	Edit	OutAsync	LOCATION_SYNC	VERSION_1	10/12/2002	Active

7. If there are no transactions, click **Add Transaction**.

   In this procedure, the node is already configured with the `LOCATION_SYNC` message.

   The Transaction Detail tab appears.



Node Name: EXTERNAL

Transaction Detail

Effective Date: 10/12/2002 *Status: Active

Transaction Type: OutAsync

Request Message: LOCATION_SYNC

Request Message Version: VERSION_1

Routing Type: Implicit

□ Override Connector

Comment:

You can add the message with which you are working.

a. Verify that the **Routing Type** is **Implicit**.

b. Click **Save**.

8. Return to the Transaction List. Perform the following steps:

   a. If there are other transactions, edit them.

   b. Set the status to **Inactive**.

      Inactive status is for initial testing only. After you test your configuration, you may change the status to Active and have as many nodes and transactions as required to satisfy your business requirements.

9. Click **Save** on the Transaction List.

You can now send XML messages to your PeopleSoft XML listener.

Configuring the TCP/IP Handler for PeopleSoft 8.1

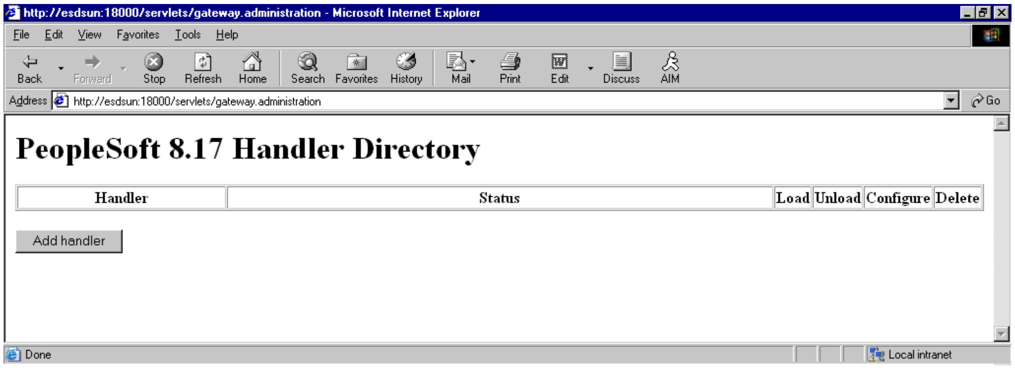
The following procedure assumes that your Application Messaging environment is properly configured and tested. See Appendix D, "Using PeopleSoft Integration Broker" for more information.

To configure the TCP/IP Handler for PeopleSoft 8.1 to send messages to Oracle Application Server:

1. In a Web browser, launch the **PeopleSoft 8.1 Gateway Configuration** servlet interface.

2. If the Simple File Handler is currently loaded, unload and delete it before proceeding.

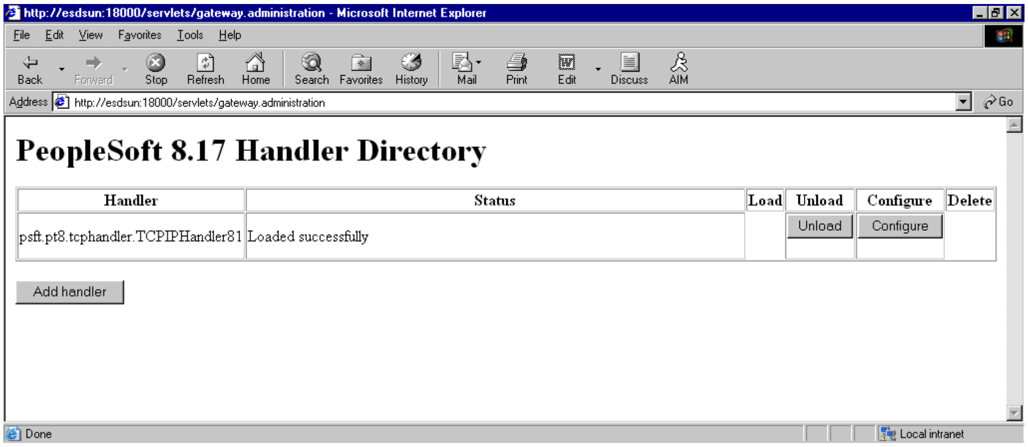
   You must see an empty Handler directory.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Add handler
Done
Local intranet

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete

3. Click **Add handler**.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete
psft.pt8.tcphandler.TCPIPHandler81	Loaded successfully		Unload	Configure	

Add handler

Perform the following steps:

a. Enter the full path of TCPIPHandler81 (case-sensitive):

```
psft.pt8.tcphandler.TCPIPHandler81
```

b. Click **Save**.

4. Click **Load**.

The PeopleSoft Handler Directory window is displayed.

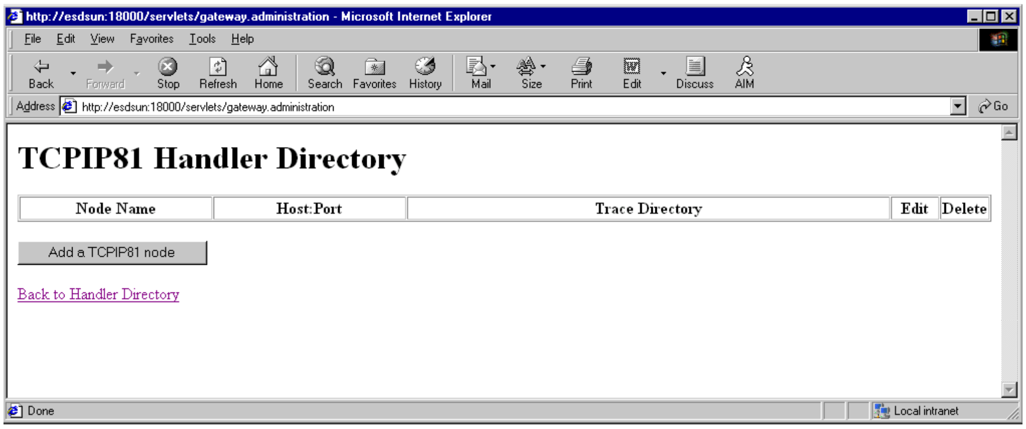
5. Click **Configure**.

The TCPIP81 Handler Directory window is displayed.

6. Click **Add a TCPIP81 node**.

> **Note:** The screens illustrating this procedure show a node named EXTERNAL. For more information about creating and using nodes, see Appendix D, "Using PeopleSoft Integration Broker" or your PeopleSoft documentation.

The Add TCPIP81 Handler window is displayed.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

TCPIP81 Handler Directory

Node Name	Host:Port	Trace Directory	Edit	Delete

Add a TCPIP81 node

Back to Handler Directory

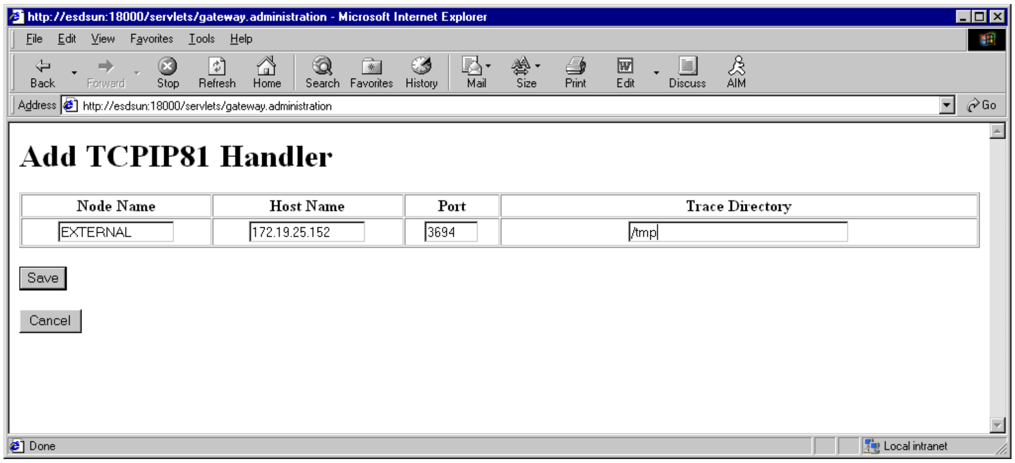
7. Enter the requested values based on the information in the following table.

Field	Value Example	Description
Node Name	EXTERNAL	The name of the TCP/IP node.
Host Name	172.19.25.152	The machine on which your PeopleSoft XML listener is listening for incoming messages.
Port	3694	The port on which your PeopleSoft XML listener is listening for incoming messages.
Trace Directory	/tmp	The directory where a trace file is created when errors occur in message delivery.

The system does not validate your entries.

8. Click **Save**.

The TCPIP81 Handler Directory window is displayed.



http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go

Add TCPIP81 Handler

Node Name	Host Name	Port	Trace Directory
EXTERNAL	172.19.25.152	3694	/tmp

Save
Cancel
Done
Local intranet

9. For your changes to take effect, click **Back to Handler Directory** to return to the PeopleSoft 8.1 Handler Directory window.

10. Click **Unload and re-Load TCPIPHandler81**.

You can now send messages from PeopleSoft to OracleAS Adapter for PeopleSoft.

Testing Your PeopleSoft Configuration

PeopleSoft 8.1 and 8.4 provide a ping node mechanism for testing your configuration. The mechanism functions identically in both versions.

Test your configuration to ensure that:

- Oracle Application Server is up and running.
- The server name and port number for PeopleSoft and Oracle Application Server match.

- The default page for HTTP exists.

To test a PeopleSoft configuration:

1. In a Web browser, open your PeopleSoft application.
2. Navigate to the message monitoring menu.

   For PeopleSoft 8.4:

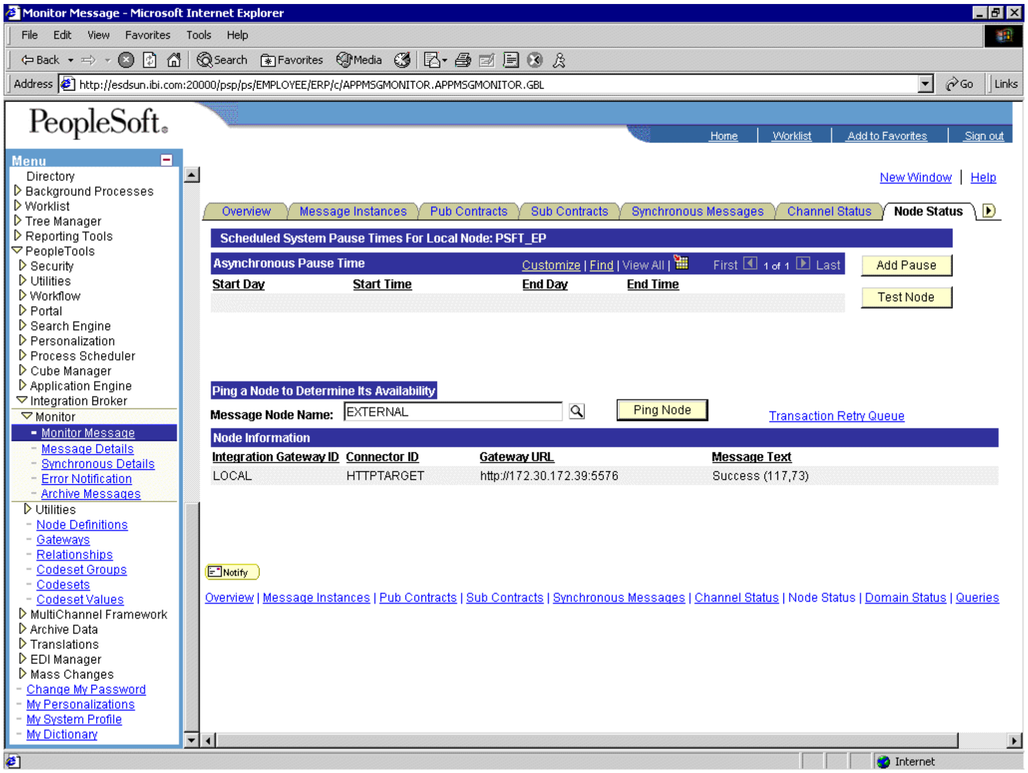
   a. In the menu pane, expand **PeopleTools**, **Integration Broker**, and **Monitor**.

   b. Select **Monitor Message**.

   For PeopleSoft 8.1:

   a. In the menu pane, expand **Home**, **PeopleTools**, **Application Message Monitor**, and **Use**.

   b. Select **Application Message Monitor**.
3. Click the **Node Status** tab.



Scheduled System Pause Times For Local Node: PSFT_EP

Asynchronous Pause Time

Start Day	Start Time	End Day	End Time

Ping a Node to Determine Its Availability

Message Node Name: EXTERNAL

Node Information

Integration Gateway ID	Connector ID	Gateway URL	Message Text
LOCAL	HTTPTARGET	http://172.30.172.39:5576	Success (117,73)

Perform the following steps:

a. From the Message Node Name list, select your node.

b. Click **Ping Node**.

If you properly configured both PeopleSoft and Oracle Application Server, you receive a Success message.

An error indicates a configuration problem. For more information, see the Integration Broker error log.

C

Using Component Interfaces

This appendix describes how to create new and modify existing component interfaces for use with OracleAS Adapter for PeopleSoft. It also describes how to apply security to those component interfaces and how to test them.

You can:

- Use component interfaces supplied by PeopleSoft with your application.
- Component interfaces also are known as Enterprise Integration Points (EIP).
- Modify an existing component interface.
- Create a new component interface.

Before using your component interface you must apply security to it and test it.

After securing and testing a component interface, you must generate its API, see Appendix A, "Generating Component Interface APIs" for more information.

> **Note:** This section is intended as a helpful supplement; it is not a substitute for PeopleSoft documentation. For complete and up-to-date information about PeopleSoft component interfaces, see the PeopleSoft Online Library for your PeopleSoft system.

Creating a Component Interface

You create component interfaces using the PeopleSoft Application Designer. For more information about Application Designer, see your PeopleSoft documentation.

Working With Properties

You can add properties from the records in the component view. You can delete a property in the component interface that you do not want to expose. You can rename properties by clicking the property and then clicking again until you can enter a new name. If you rename a property, it can be referenced in the component interface only by the new name, not by the underlying component name.

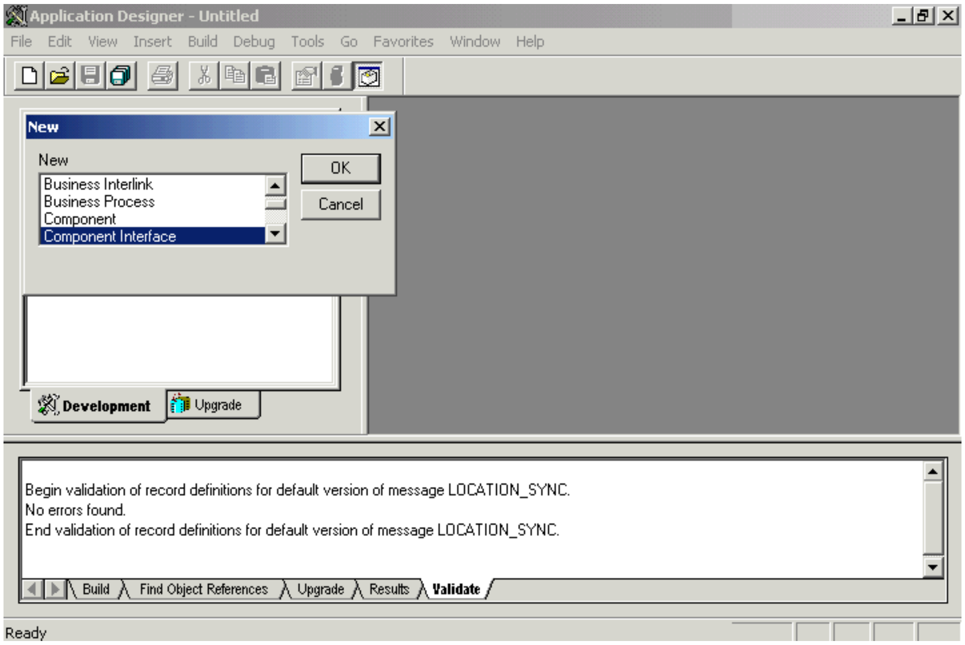
Properties may have various icons adjacent to them. For example, EMPLID has an icon indicating that it is a key field from the underlying record. NAME has an icon indicating that it is an alternate key field from the underlying record. For a complete list of property icons, see the PeopleBooks documentation.

Creating a New Component Interface

To create a component interface:

1. Open the PeopleSoft Application Designer.
2. Select **New** from the **File** menu.

   The New dialog box is displayed.


Application Designer - Untitled
File Edit View Insert Build Debug Tools Go Favorites Window Help
New
OK
Cancel
Development
Upgrade
Build
Find Object References
Upgrade
Results
Validate

New

- Business Interlink
- Business Process
- Component
- Component Interface

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

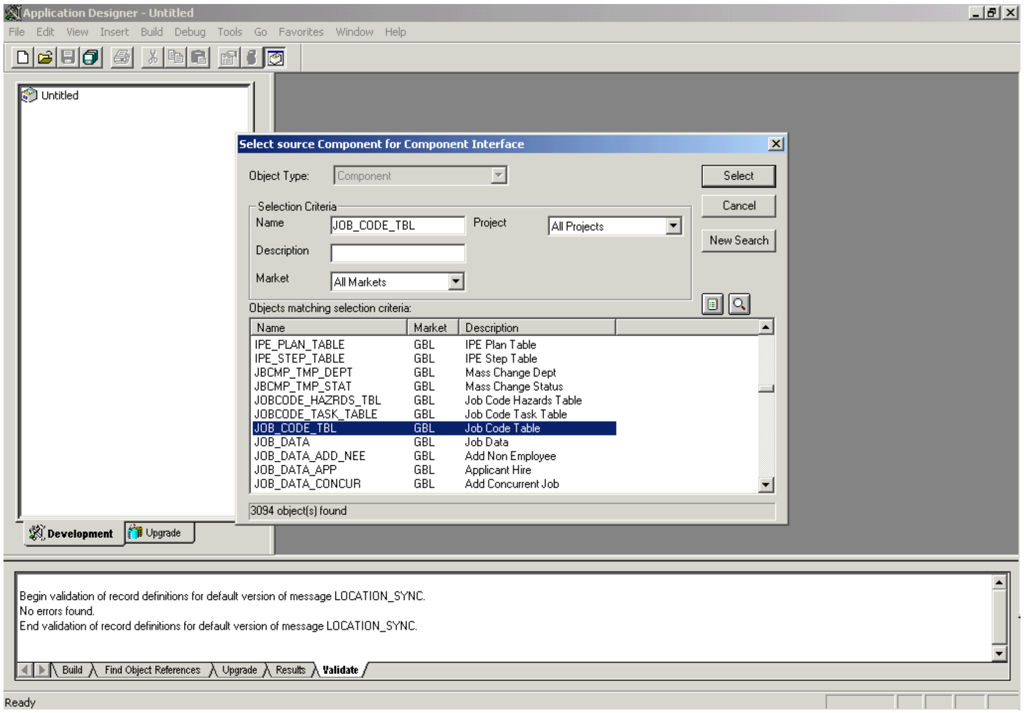
Ready

Perform the following steps:

a. Select **Component Interface**.

b. Click **OK**.

   The Select Source Component for Component Interface dialog box is displayed.


Application Designer - Untitled
File Edit View Insert Build Debug Tools Go Favorites Window Help
Untitled
Select source Component for Component Interface
Select
Cancel
New Search
Development
Upgrade
Build
Find Object References
Upgrade
Results
Validate

Object Type: Component

Selection Criteria

Name: JOB_CODE_TBL

Project: All Projects

Description:

Market: All Markets

Objects matching selection criteria:

Name	Market	Description
IPE_PLAN_TABLE	GBL	IPE Plan Table
IPE_STEP_TABLE	GBL	IPE Step Table
JBCMP_TMP_DEPT	GBL	Mass Change Dept
JBCMP_TMP_STAT	GBL	Mass Change Status
JOBCODE_HAZRDS_TBL	GBL	Job Code Hazards Table
JOBCODE_TASK_TABLE	GBL	Job Code Task Table
JOB_CODE_TBL	GBL	Job Code Table
JOB_DATA	GBL	Job Data
JOB_DATA_ADD_NEE	GBL	Add Non Employee
JOB_DATA_APP	GBL	Applicant Hire
JOB_DATA_CONCUR	GBL	Add Concurrent Job

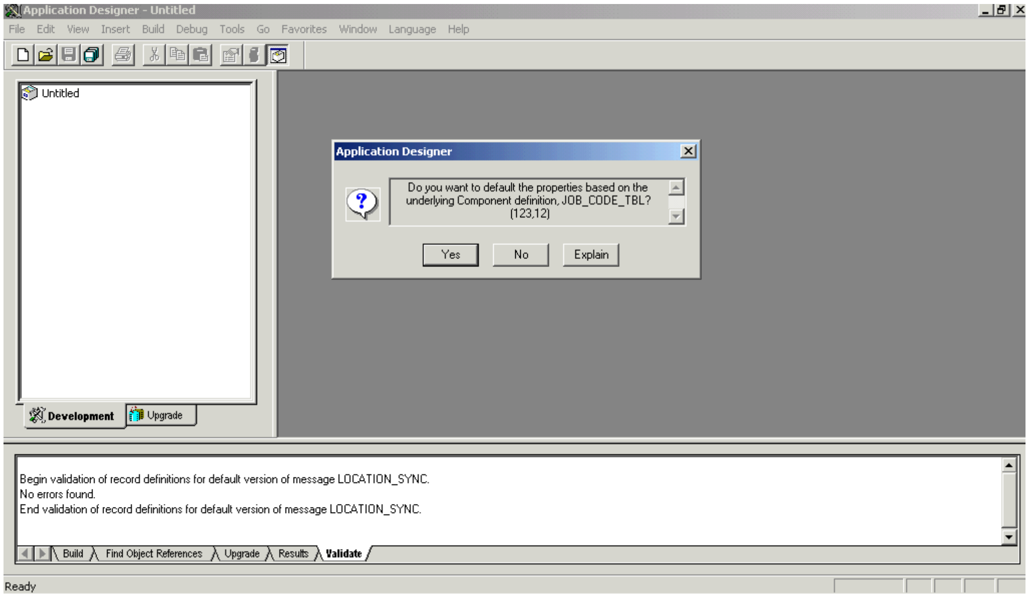
3094 object(s) found

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

Ready

3. Highlight the component to use as a basis for the component interface and click **Select**.

   The Application Designer dialog box is displayed.



Do you want to default the properties based on the underlying Component definition, JOB_CODE_TBL? (123,12)

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

Note: If the component interface is large, expose the component properties manually.

---

4. To create the component interface without displaying properties and to expose component properties manually, click **No**.

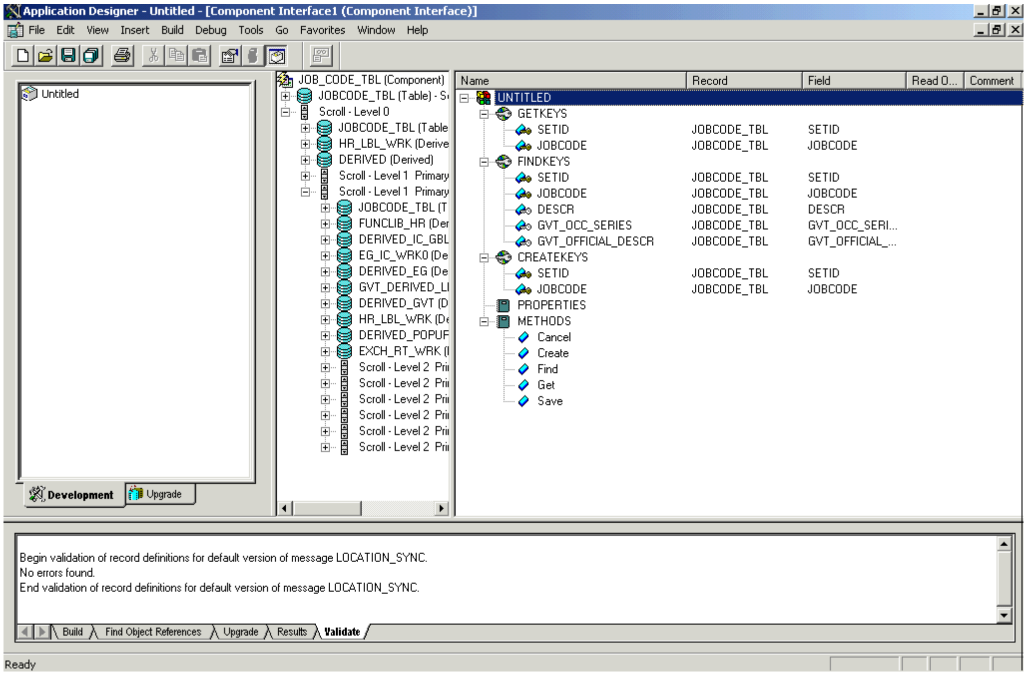
   Perform the following steps:

   a. Drag the relevant fields from the left pane to the right pane.

   b. To select various functions to perform, right-click either the right or left pane, depending on which pane is active.

      For a complete list of functions, see the PeopleBooks documentation.

5. To create the component interface and display the properties of the underlying component interface, click **Yes**.



Name	Record	Field	Read O...	Comment
UNTITLED				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
DESCR	JOBCODE_TBL	DESCR		
GVT_OCC_SERIES	JOBCODE_TBL	GVT_OCC_SERI...		
GVT_OFFICIAL_DESCR	JOBCODE_TBL	GVT_OFFICIAL_...		
CREATEKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
PROPERTIES				
METHODS				
Cancel				
Create				
Find				
Get				
Save				

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

Standard Methods

The standard methods for the component interface are:

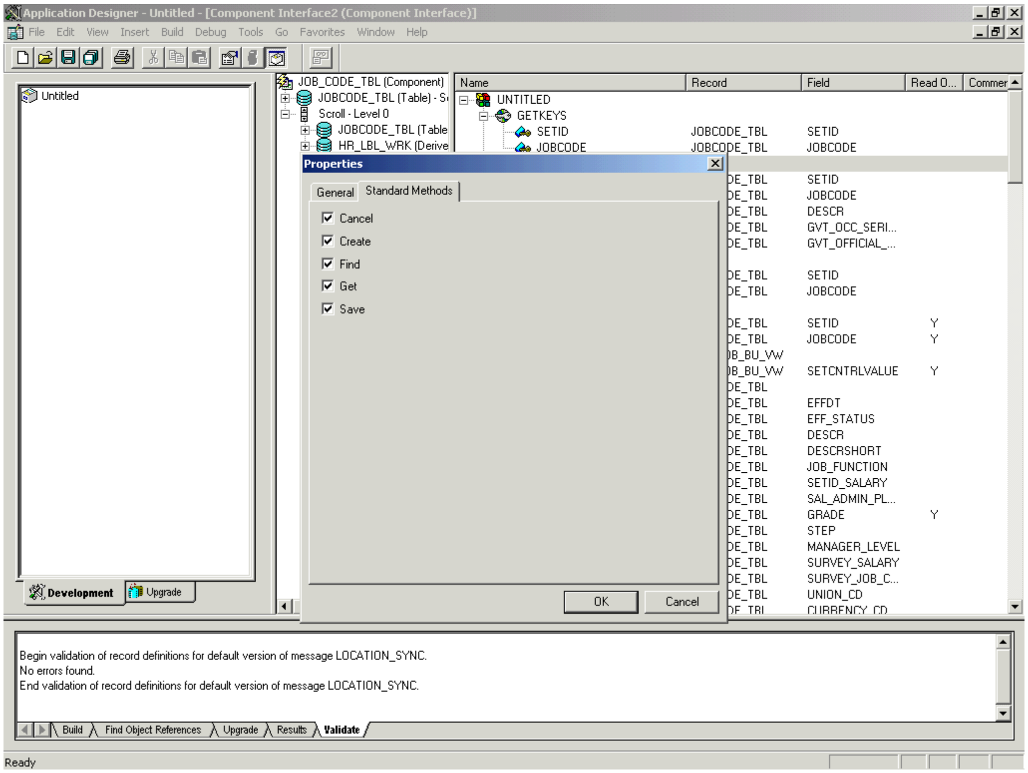
- Create
- Find
- Get
- Save

Only those methods in the underlying component are available. For example, if the underlying component does not contain Add capabilities, Create is not available.

Viewing or Changing Available Methods

To view or change available methods:

1. Display the Component Interface Properties dialog box.



☑ Cancel
☑ Create
☑ Find
☑ Get
☑ Save

Name	Record	Field	Read O...	Commer
UNTITLED				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
	DE_TBL	SETID		
	DE_TBL	JOBCODE		
	DE_TBL	DESCR		
	DE_TBL	GVT_OCC_SERI...		
	DE_TBL	GVT_OFFICIAL_...		
	DE_TBL	SETID		
	DE_TBL	JOBCODE		
	DE_TBL	SETID	Y	
	DE_TBL	JOBCODE	Y	
	B_BU_VW			
	B_BU_VW	SETCNTRLVALUE	Y	
	DE_TBL			
	DE_TBL	EFFDT		
	DE_TBL	EFF_STATUS		
	DE_TBL	DESCR		
	DE_TBL	DESCRSHORT		
	DE_TBL	JOB_FUNCTION		
	DE_TBL	SETID_SALARY		
	DE_TBL	SAL_ADMIN_PL...		
	DE_TBL	GRADE	Y	
	DE_TBL	STEP		
	DE_TBL	MANAGER_LEVEL		
	DE_TBL	SURVEY_SALARY		
	DE_TBL	SURVEY_JOB_C...		
	DE_TBL	UNION_CD		
	DE_TBL	CURRENCY_CD		

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

2. Click the **Standard Methods** tab.
3. Select the desired methods.

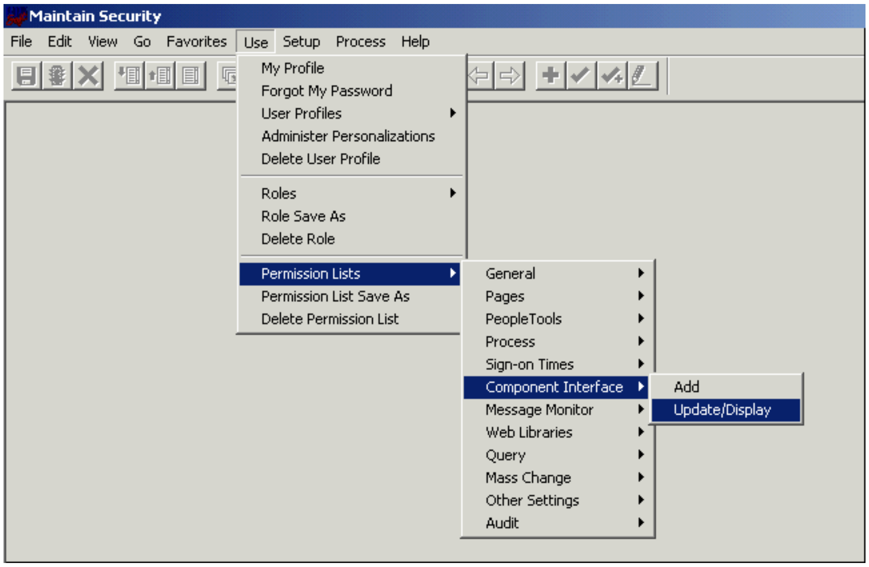
Securing a Component Interface

You must set up security for the component interface before you can begin testing.

Configuring Component Interface Security for PeopleSoft Version 8.1x

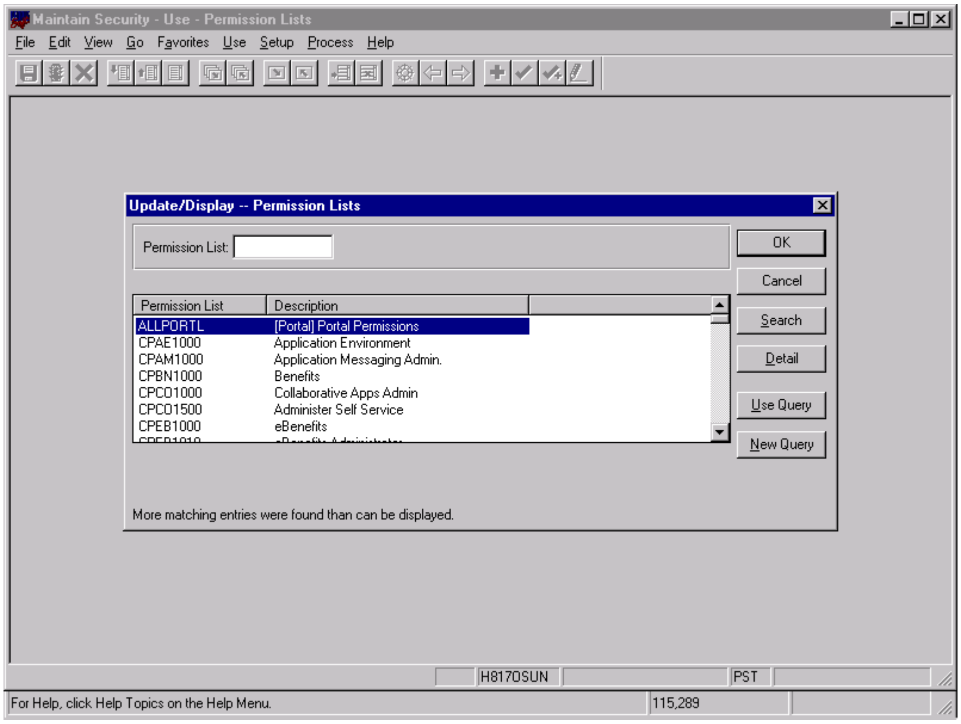
The following procedure describes how to configure component interface security for PeopleSoft Version 8.1 in 2- and 3-tier mode.

To configure component interface security:



1. From the **Use** menu, select **Permission Lists**, **Component Interface**, and then click **Update/Display**.

   The Permission Lists dialog box is displayed.


Maintain Security - Use - Permission Lists
File Edit View Go Favorites Use Setup Process Help
Update/Display -- Permission Lists
OK
Cancel
Search
Detail
Use Query
New Query
H8170SUN
PST
For Help, click Help Topics on the Help Menu.
115,289

Permission List:

Permission List	Description
ALLPORTL	[Portal] Portal Permissions
CPAE1000	Application Environment
CPAM1000	Application Messaging Admin.
CPBN1000	Benefits
CPCO1000	Collaborative Apps Admin
CPCO1500	Administer Self Service
CPEB1000	eBenefits

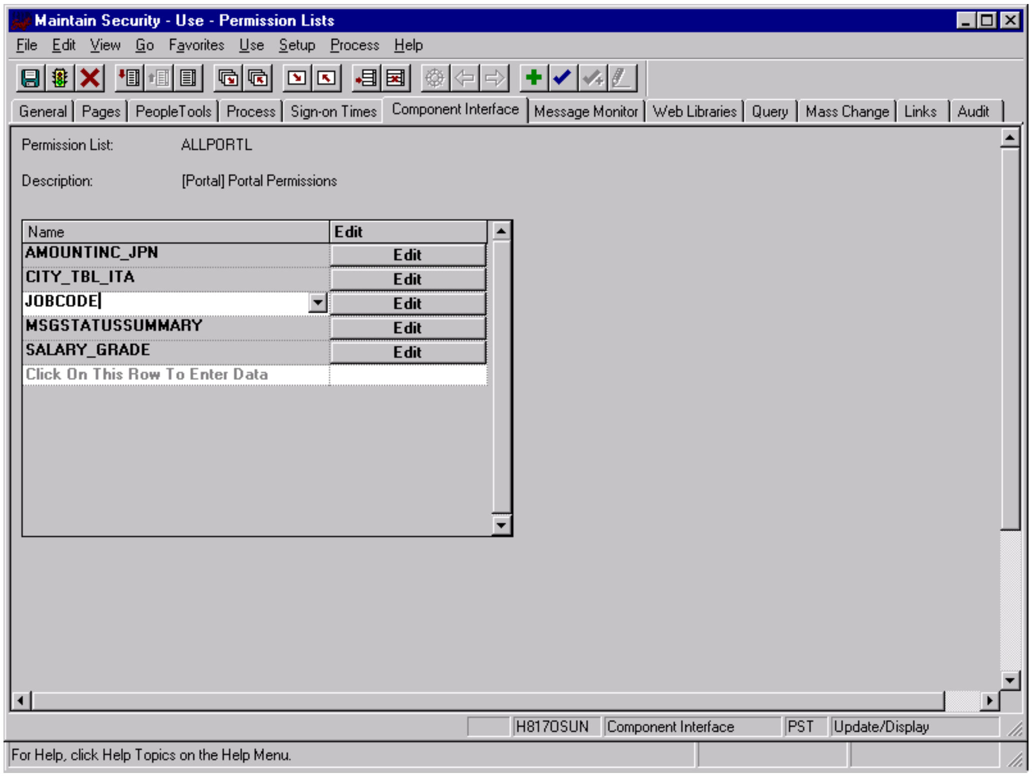
More matching entries were found than can be displayed.

Before Security can be set, you must identify the permission lists.

2. Select the relevant permission list and click **OK**.

   For more information on permission lists, see the PeopleBooks documentation.

   The following pane is displayed.



Permission List: ALLPORTL

Description: [Portal] Portal Permissions

Name	Edit
AMOUNTINC_JPN	Edit
CITY_TBL_ITA	Edit
JOBCODE	Edit
MSGSTATUSSUMMARY	Edit
SALARY_GRADE	Edit
Click On This Row To Enter Data	

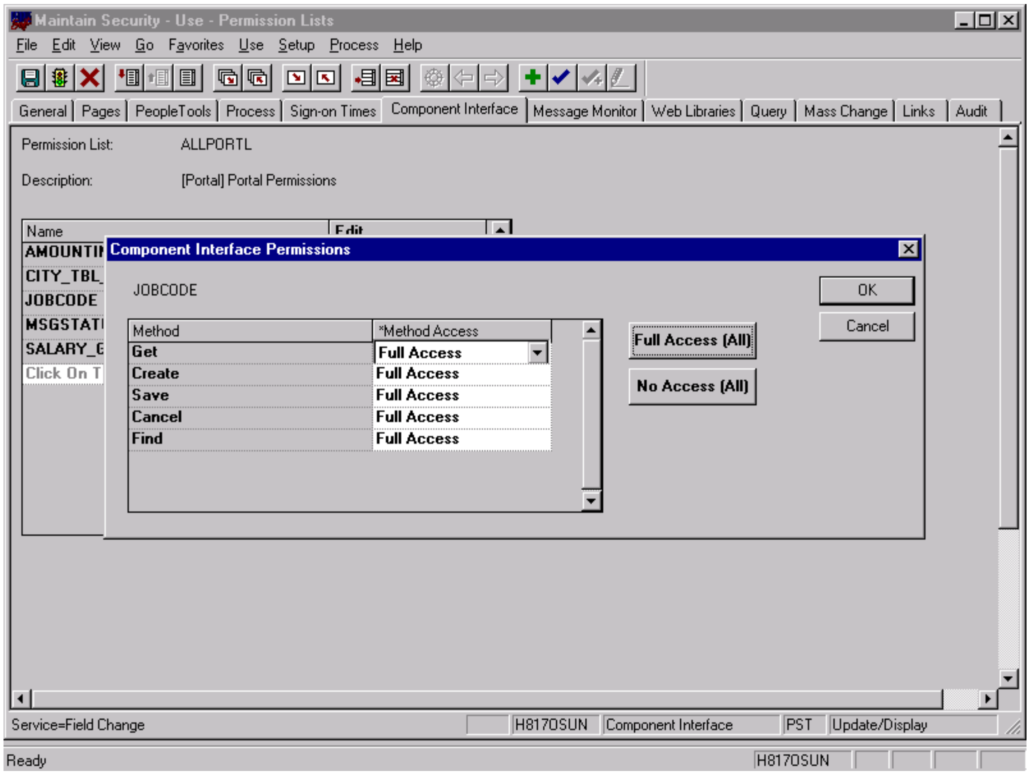
H8170SUN | Component Interface | PST | Update/Display

For Help, click Help Topics on the Help Menu.

3. Insert the new component interface that you created.
4. Click **Edit**.

   When you select the component interface, all available methods appear, including user-defined methods. This enables you to specify whether this particular Permission List must have full or partial access.

   In the following example, the ALLPORTL Permission List has full access to all methods.


Maintain Security - Use - Permission Lists
File Edit View Go Favorites Use Setup Process Help
General Pages PeopleTools Process Sign-on Times Component Interface Message Monitor Web Libraries Query Mass Change Links Audit
Component Interface Permissions
OK
Cancel
Full Access (All)
No Access (All)
Service=Field Change H8170SUN Component Interface PST Update/Display
Ready H8170SUN

Permission List: ALLPORTL

Description: [Portal] Portal Permissions

Name	Edit
AMOUNTIN	
CITY_TBL_	
JOBCODE	
MSGSTATI	
SALARY_G	
Click On T	

JOBCODE

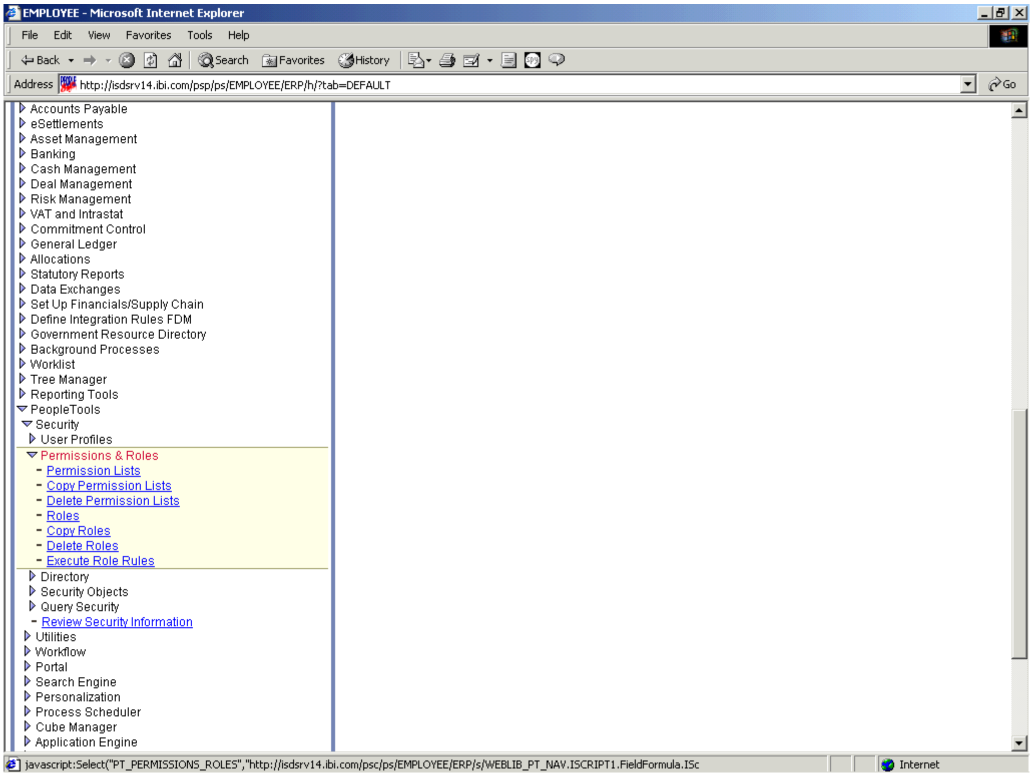
Method	*Method Access
Get	Full Access
Create	Full Access
Save	Full Access
Cancel	Full Access
Find	Full Access

5. Select the desired level of access.
6. Click **OK**.

Configuring Component Interface Security for PeopleSoft Version 8.4 or Higher

The following procedure describes how to configure component interface security for PeopleSoft Version 8.4 or higher.

To configure interface security:



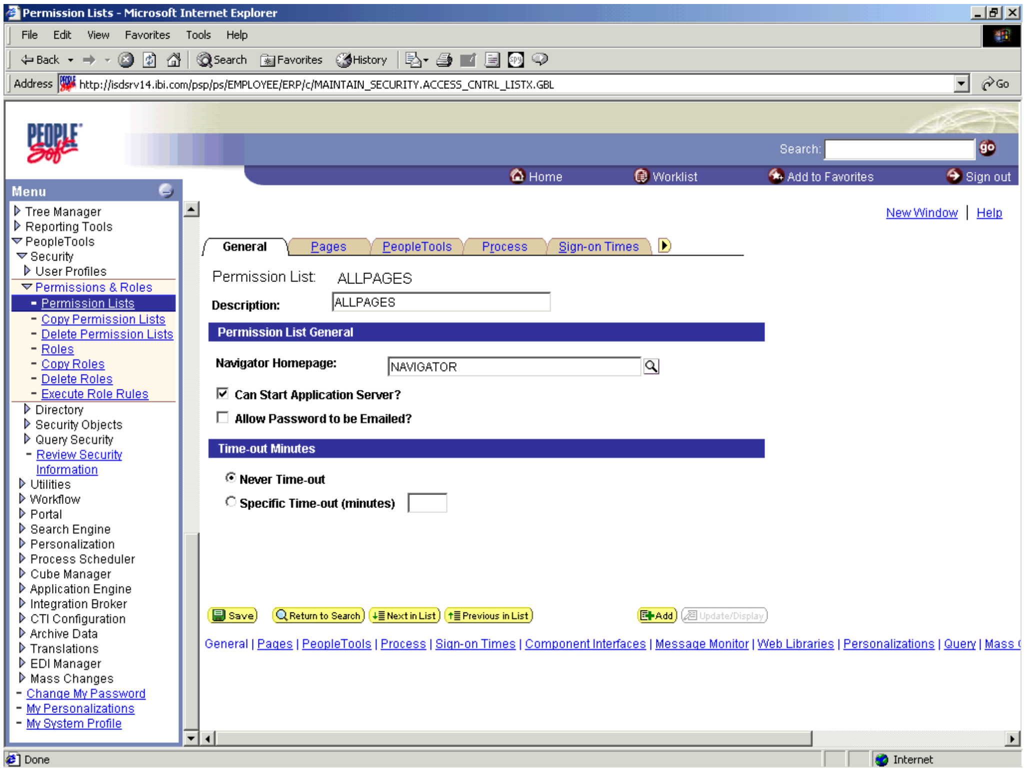
1. Expand **PeopleTools**, **Security**, **User Profiles**, and **Permissions & Roles** and then click **Permission Lists**.

2. Click **Search**.

   The Permission Lists Search pane is displayed.

3. Select the relevant permission list.

   The following pane is displayed.



Permission List: ALLPAGES

Description: ALLPAGES

Navigator Homepage: NAVIGATOR

☑ Can Start Application Server?

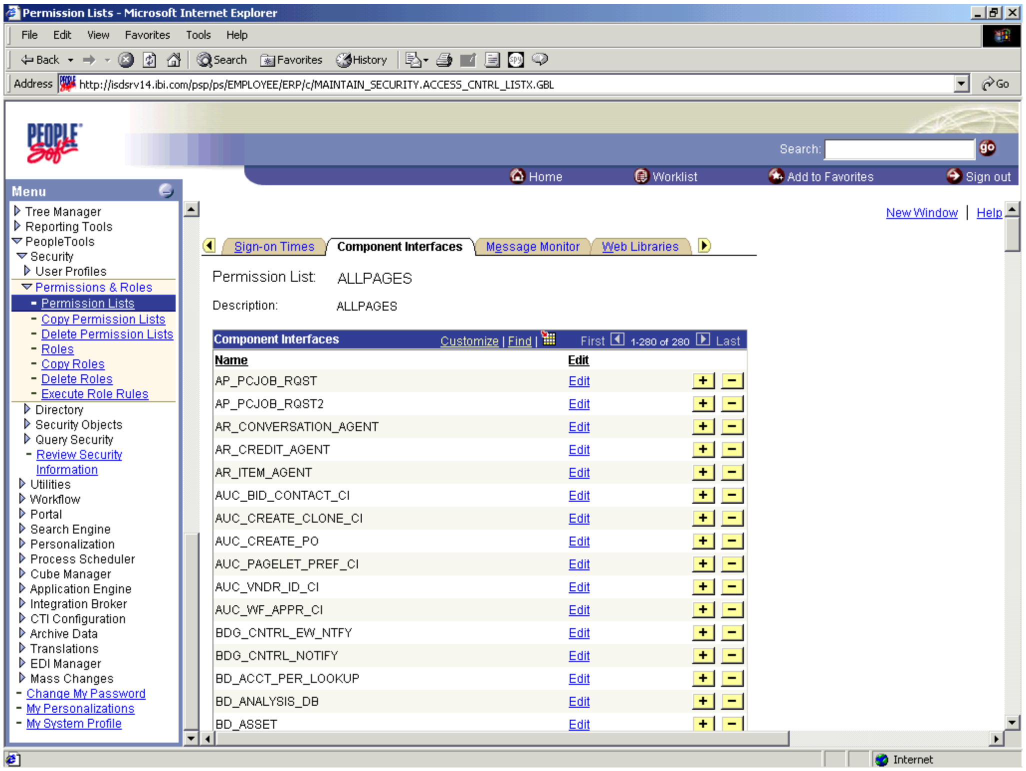
□ Allow Password to be Emailed?

● Never Time-out

○ Specific Time-out (minutes)

4. Click the right arrow next to the **Sign-on Times** tab.

The **Component Interfaces** tab appears.



Permission List: ALLPAGES

Description: ALLPAGES

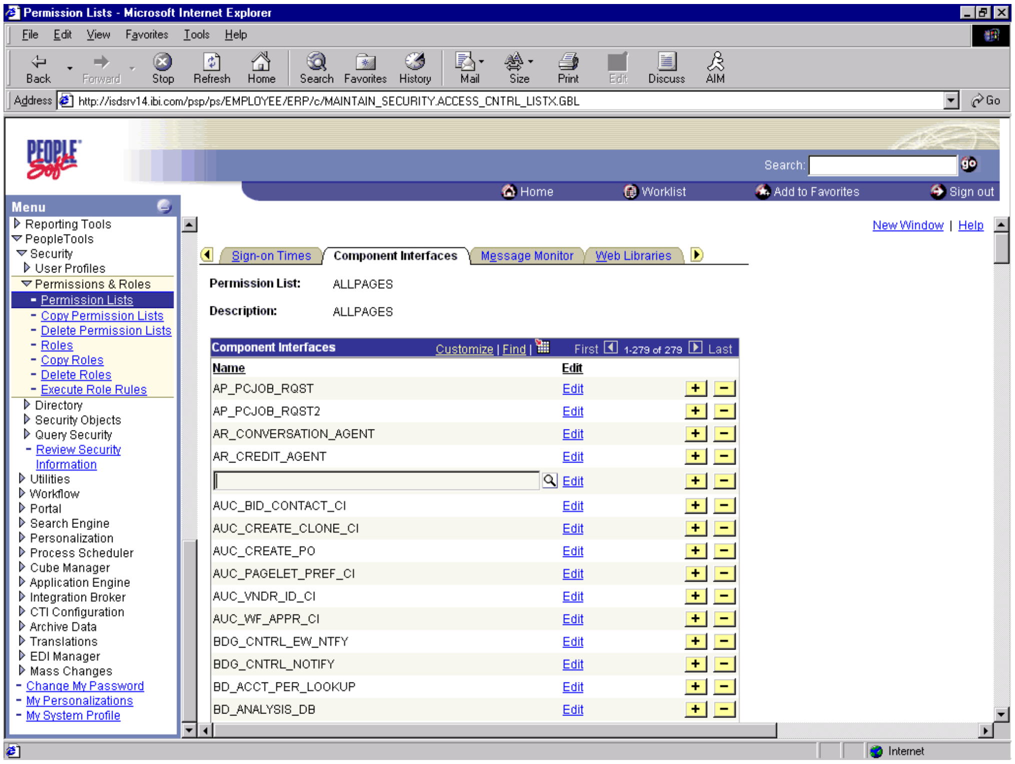
Component Interfaces — Customize | Find | First 1-280 of 280 Last

Name	Edit		
AP_PCJOB_RQST	Edit	+	−
AP_PCJOB_RQST2	Edit	+	−
AR_CONVERSATION_AGENT	Edit	+	−
AR_CREDIT_AGENT	Edit	+	−
AR_ITEM_AGENT	Edit	+	−
AUC_BID_CONTACT_CI	Edit	+	−
AUC_CREATE_CLONE_CI	Edit	+	−
AUC_CREATE_PO	Edit	+	−
AUC_PAGELET_PREF_CI	Edit	+	−
AUC_VNDR_ID_CI	Edit	+	−
AUC_WF_APPR_CI	Edit	+	−
BDG_CNTRL_EW_NTFY	Edit	+	−
BDG_CNTRL_NOTIFY	Edit	+	−
BD_ACCT_PER_LOOKUP	Edit	+	−
BD_ANALYSIS_DB	Edit	+	−
BD_ASSET	Edit	+	−

5. Click the **Component Interfaces** tab.

6. Click + to add a new row to the Component Interfaces list.

   A field appears where you can enter the component interface name.



Permission List: ALLPAGES

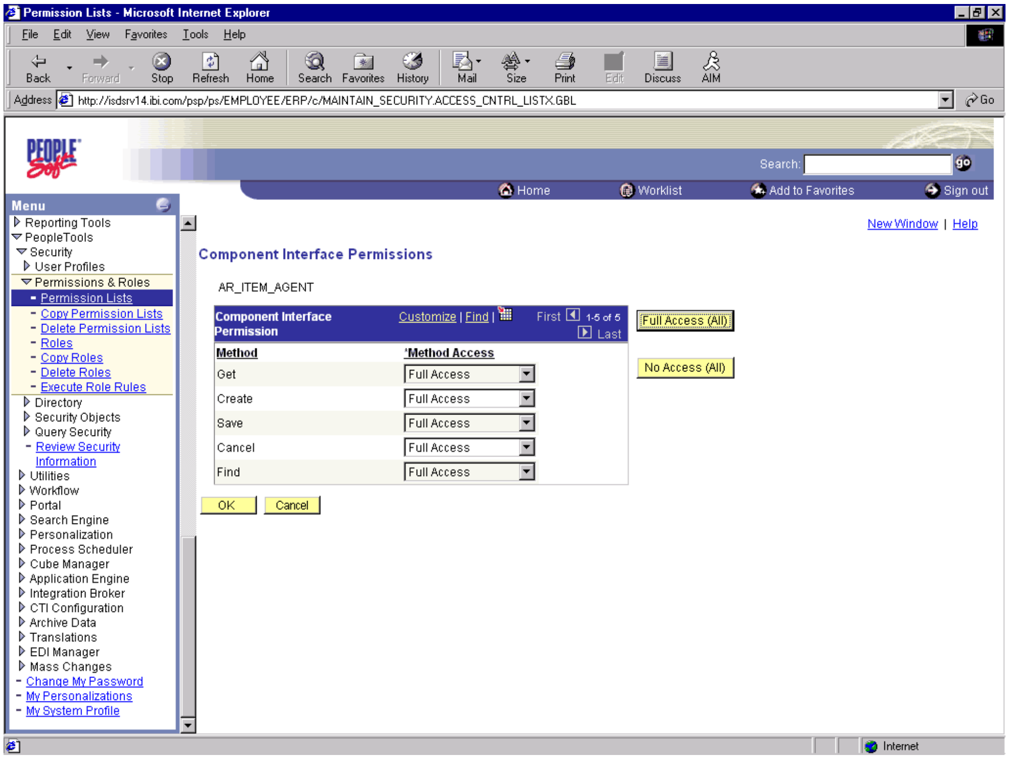
Description: ALLPAGES

Component Interfaces — Customize | Find | First 1-279 of 279 Last

Name	Edit		
AP_PCJOB_RQST	Edit	+	−
AP_PCJOB_RQST2	Edit	+	−
AR_CONVERSATION_AGENT	Edit	+	−
AR_CREDIT_AGENT	Edit	+	−
	Edit	+	−
AUC_BID_CONTACT_CI	Edit	+	−
AUC_CREATE_CLONE_CI	Edit	+	−
AUC_CREATE_PO	Edit	+	−
AUC_PAGELET_PREF_CI	Edit	+	−
AUC_VNDR_ID_CI	Edit	+	−
AUC_WF_APPR_CI	Edit	+	−
BDG_CNTRL_EW_NTFY	Edit	+	−
BDG_CNTRL_NOTIFY	Edit	+	−
BD_ACCT_PER_LOOKUP	Edit	+	−
BD_ANALYSIS_DB	Edit	+	−

7. Enter the component interface name and click **Edit**.

   This example uses the component interface AR_ITEM_AGENT.



Component Interface Permissions

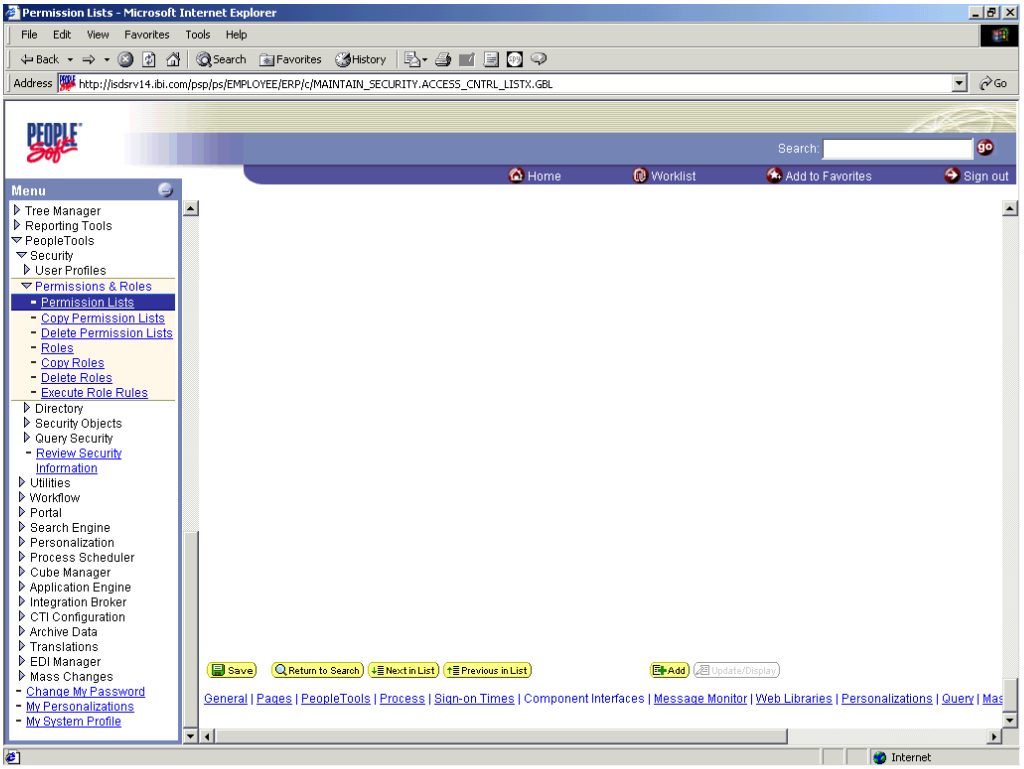
AR_ITEM_AGENT

Component Interface Permission — Customize | Find | First 1-5 of 5 Last

Method	*Method Access
Get	Full Access
Create	Full Access
Save	Full Access
Cancel	Full Access
Find	Full Access

8. From the lists, select the desired access level for each method.
9. Click **OK**.

   The following pane is displayed.



10. Scroll down in the right pane and click **Save**.

Testing a Component Interface

OracleAS Adapter for PeopleSoft uses PeopleSoft metadata and component interfaces; therefore, it can accommodate new or modified component interfaces. The adapter makes no assumptions about component interfaces except that they are logical and valid. Each component interface must be tested before being used as a source for the adapter.

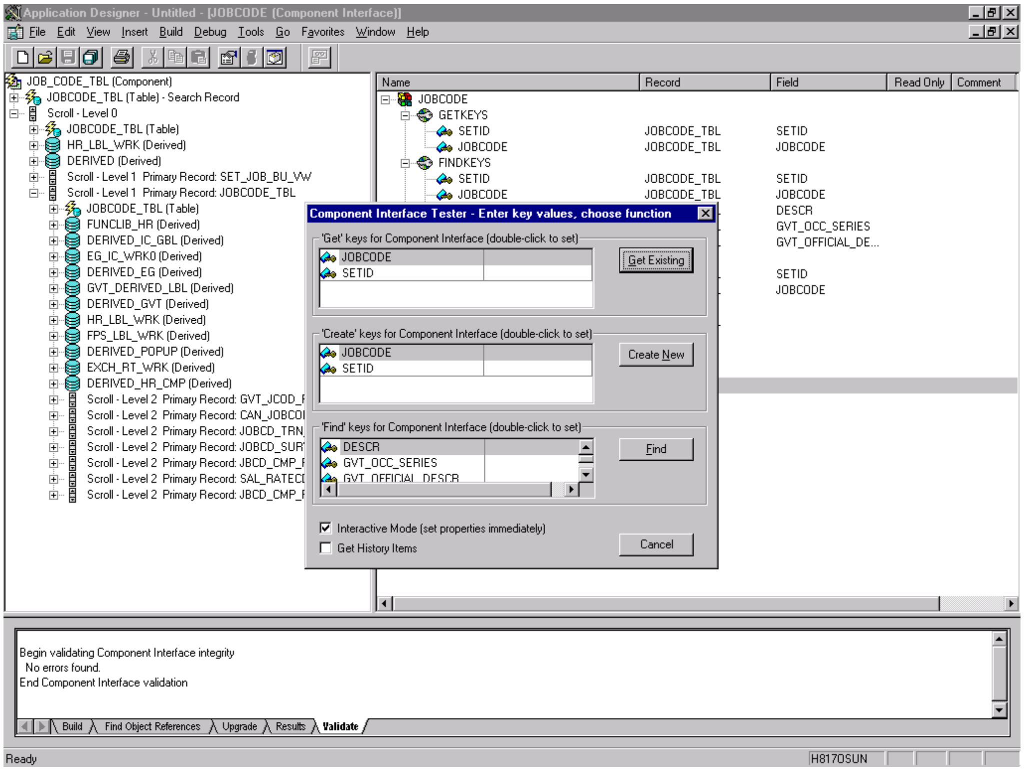
If changes are made to the underlying application by the user or by a PeopleSoft upgrade and the changes invalidate a component interface, the user must repair the invalid component interface before the adapter uses it.

Testing a Component Interface

To test a component interface:

1. In Application Designer, select **Test Component Interface** from the **Tools** menu.

   The Component Interface Tester dialog box is displayed.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_BU_VW
Scroll - Level 1 Primary Record: JOBCODE_TBL
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JCOD_
Scroll - Level 2 Primary Record: CAN_JOBCO
Scroll - Level 2 Primary Record: JOBCD_TRN
Scroll - Level 2 Primary Record: JOBCD_SUR
Scroll - Level 2 Primary Record: JBCD_CMP_
Scroll - Level 2 Primary Record: SAL_RATEC
Scroll - Level 2 Primary Record: JBCD_CMP_
Component Interface Tester - Enter key values, choose function
'Get' keys for Component Interface (double-click to set)
JOBCODE
SETID
Get Existing
'Create' keys for Component Interface (double-click to set)
JOBCODE
SETID
Create New
'Find' keys for Component Interface (double-click to set)
DESCR
GVT_OCC_SERIES
GVT_OFFICIAL_DESCR
Find
Cancel
Build
Find Object References
Upgrade
Results
Validate
Ready
H8170SUN

Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
		DESCR		
		GVT_OCC_SERIES		
		GVT_OFFICIAL_DE...		
		SETID		
		JOBCODE		

☑ Interactive Mode (set properties immediately)

☐ Get History Items

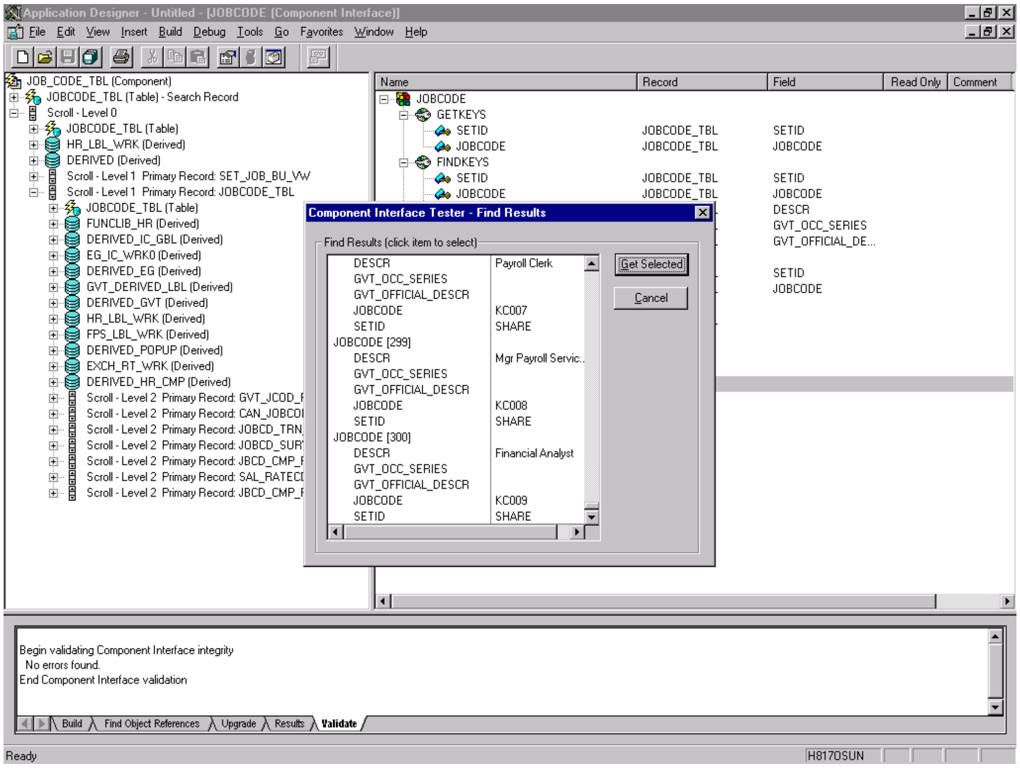
```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

2. If required, click the Component Interface Tester dialog box to bring it to the foreground.

3. To test the component interface, use one the following methods.

   - To test the component interface using the Find method, click **Find**.

     The Component Interface Tester - Find Results dialog box displays all of the possible entries for the underlying component. If there are more than 300 entries, a message appears.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_BU_VW
Scroll - Level 1 Primary Record: JOBCODE_TBL
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JCOD_
Scroll - Level 2 Primary Record: CAN_JOBCO
Scroll - Level 2 Primary Record: JOBCD_TRN
Scroll - Level 2 Primary Record: JOBCD_SUR
Scroll - Level 2 Primary Record: JBCD_CMP_
Scroll - Level 2 Primary Record: SAL_RATECI
Scroll - Level 2 Primary Record: JBCD_CMP_
Component Interface Tester - Find Results
Find Results (click item to select)
Get Selected
Cancel
Build
Find Object References
Upgrade
Results
Validate
Ready
H817OSUN

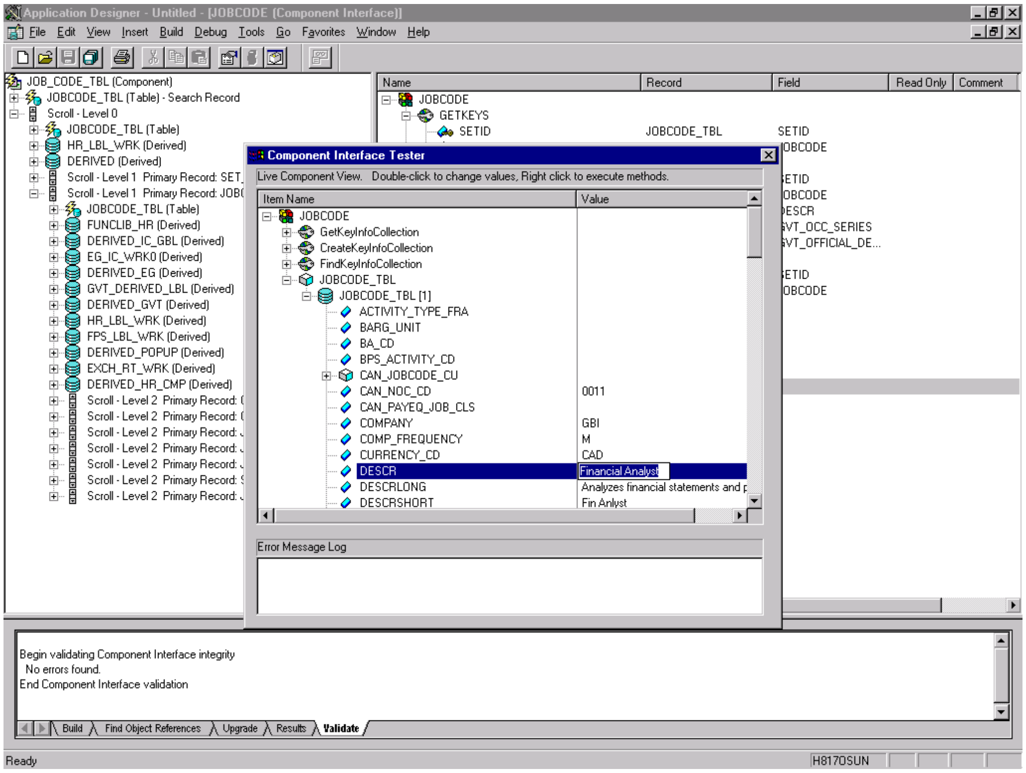
Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
		DESCR		
		GVT_OCC_SERIES		
		GVT_OFFICIAL_DE...		
		SETID		
		JOBCODE		

DESCR	Payroll Clerk
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	
JOBCODE	KC007
SETID	SHARE
JOBCODE [299]	
DESCR	Mgr Payroll Servic..
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	
JOBCODE	KC008
SETID	SHARE
JOBCODE [300]	
DESCR	Financial Analyst
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	
JOBCODE	KC009
SETID	SHARE

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

- In the left pane of the Find Results dialog box, select a field.
- To display the relevant data for that particular field, click **Get Selected**.

  The following dialog box is displayed.



Live Component View. Double-click to change values, Right click to execute methods.

Item Name	Value
JOBCODE	
GetKeyInfoCollection	
CreateKeyInfoCollection	
FindKeyInfoCollection	
JOBCODE_TBL	
JOBCODE_TBL [1]	
ACTIVITY_TYPE_FRA	
BARG_UNIT	
BA_CD	
BPS_ACTIVITY_CD	
CAN_JOBCODE_CU	
CAN_NOC_CD	0011
CAN_PAYEQ_JOB_CLS	
COMPANY	GBI
COMP_FREQUENCY	M
CURRENCY_CD	CAD
DESCR	Financial Analyst
DESCRLONG	Analyzes financial statements and p
DESCRSHORT	Fin Anlyst

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

- If the security settings permit, you can change the values in the individual fields.

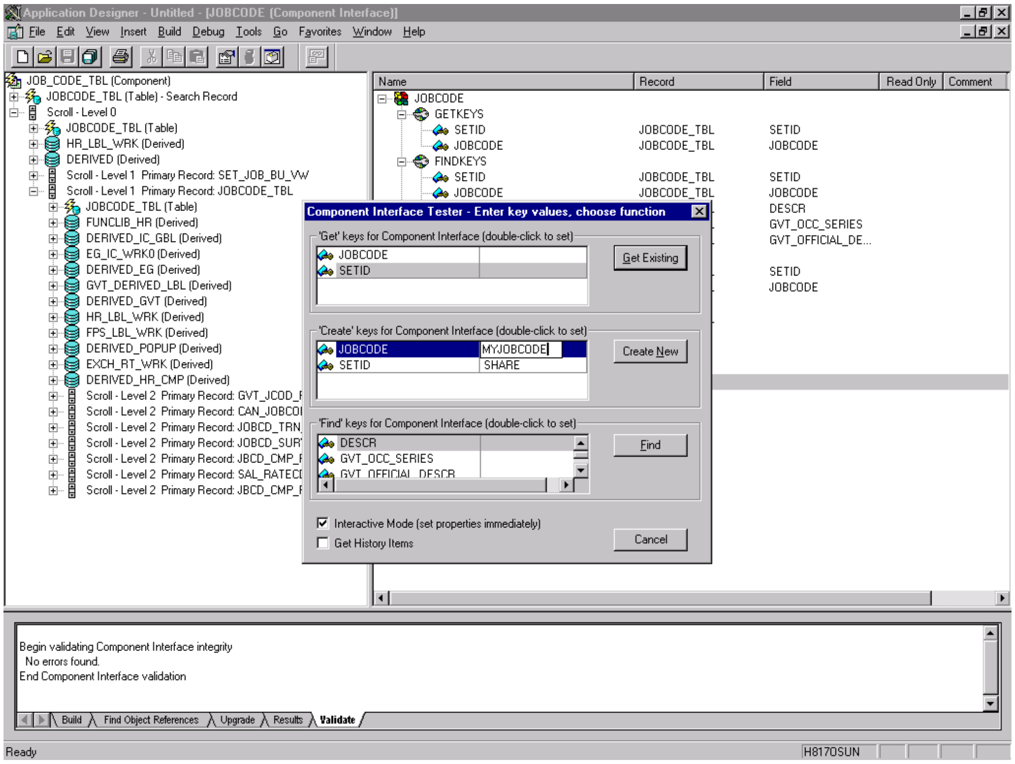
4. To test the component interface using the Get method, perform the following steps:

   a. Enter the existing key(s).

   b. Click **Get Existing**.

      This returns the exposed properties for the key that you entered. You can change values if Update access was specified.

   Alternatively, you can test using the Create method.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_BU_VW
Scroll - Level 1 Primary Record: JOBCODE_TBL
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JCOD_F
Scroll - Level 2 Primary Record: CAN_JOBCOI
Scroll - Level 2 Primary Record: JOBCD_TRN
Scroll - Level 2 Primary Record: JOBCD_SUR
Scroll - Level 2 Primary Record: JBCD_CMP_F
Scroll - Level 2 Primary Record: SAL_RATECI
Scroll - Level 2 Primary Record: JBCD_CMP_F
Get Existing
Create New
Find
Cancel
Build
Find Object References
Upgrade
Results
Validate
Ready
H8170SUN

Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
		DESCR		
		GVT_OCC_SERIES		
		GVT_OFFICIAL_DE...		
		SETID		
		JOBCODE		

Component Interface Tester - Enter key values, choose function

'Get' keys for Component Interface (double-click to set)

JOBCODE	
SETID	

'Create' keys for Component Interface (double-click to set)

JOBCODE	MYJOBCODE
SETID	SHARE

'Find' keys for Component Interface (double-click to set)

DESCR	
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	

☑ Interactive Mode (set properties immediately)

☐ Get History Items

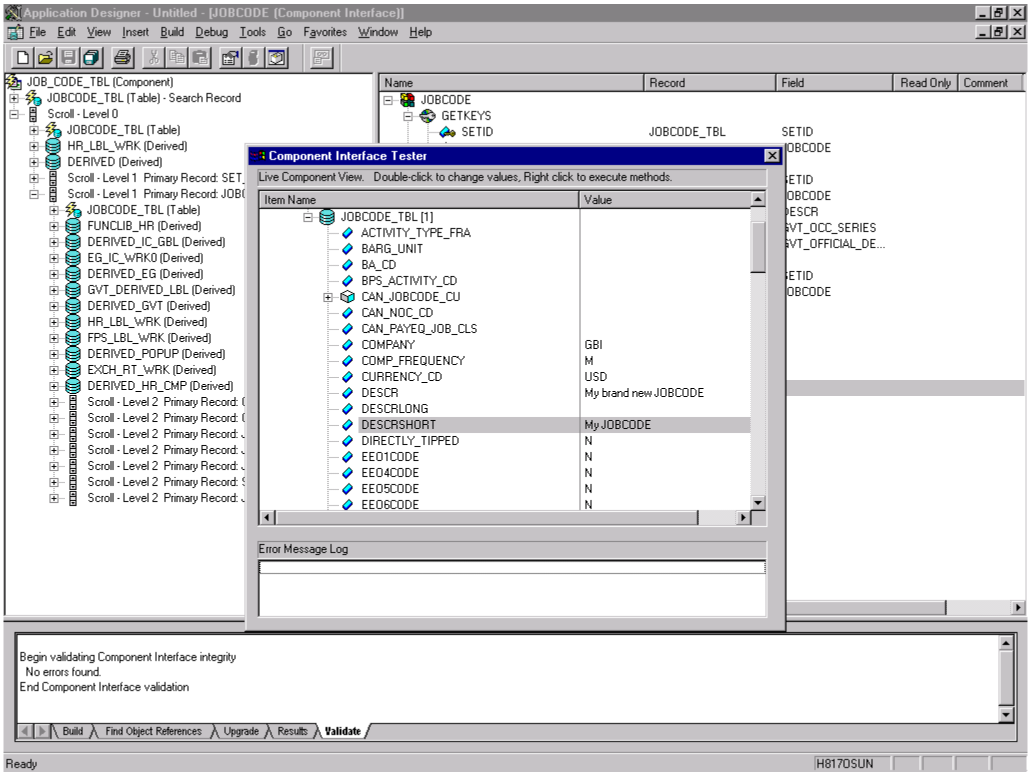
```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

Perform the following steps:

a. Enter all required key values.

b. Click **Create New**.

   When you enter valid values in Create keys, a pane showing the JOBCODE data is displayed after the Table name is expanded with default data in place.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET
Scroll - Level 1 Primary Record: JOB
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record:
Name Record Field Read Only Comment
JOBCODE
GETKEYS
SETID JOBCODE_TBL SETID
Component Interface Tester
Live Component View. Double-click to change values, Right click to execute methods.
Error Message Log
Build Find Object References Upgrade Results Validate
Ready H8170SUN

Item Name	Value
JOBCODE_TBL [1]	
ACTIVITY_TYPE_FRA	
BARG_UNIT	
BA_CD	
BPS_ACTIVITY_CD	
CAN_JOBCODE_CU	
CAN_NOC_CD	
CAN_PAYEQ_JOB_CLS	
COMPANY	GBI
COMP_FREQUENCY	M
CURRENCY_CD	USD
DESCR	My brand new JOBCODE
DESCRLONG	
DESCRSHORT	My JOBCODE
DIRECTLY_TIPPED	N
EEO1CODE	N
EEO4CODE	N
EEO5CODE	N
EEO6CODE	N

```
Begin validating Component Interface integrity
  No errors found.
End Component Interface validation
```

You can change fields at this point. Changes are validated against the component's underlying business logic.

c. After you finish making changes, right-click the top item in the pane.

5. To save your changes, click **Save**.

   The keys used to create the record can be used with the Get method for viewing data. The data that was added can be viewed in the PeopleSoft Component as shown in the following example. The Effective Date is one of the default values.

You have finished testing the component interface. Before using the component interface, you must generate its API. See Appendix A, "Generating Component Interface APIs" for more information.

D

Using PeopleSoft Integration Broker

This appendix describes how to configure and test PeopleSoft Integration Broker (release 8.4) and PeopleSoft Application Messaging (release 8.1) using a PeopleSoft-supplied File Output interface. In PeopleSoft release 8.1, the messaging architecture is called Application Messaging and includes Application Messaging Gateway. In release 8.4, the messaging architecture is called Integration Broker, which includes Integration Gateway. When discussing release-generic issues, this section uses release 8.4 terminology. When discussing release-specific issues, it uses release-specific terminology.

> **Note:** This section is not a substitute for PeopleSoft documentation. For more complete and up-to-date information on PeopleSoft Messaging and Integration Broker, see the *PeopleSoft Online Library* for your PeopleSoft system.

PeopleSoft Integration Broker

PeopleSoft Integration Broker provides a mechanism for communicating with the outside world using XML files. Communication can take place between different PeopleSoft applications or between PeopleSoft and third-party systems.

To subscribe to data, third-party applications can accept and process XML messages posted by PeopleSoft using the available PeopleSoft connectors or by adding a custom built connector to the Integration Gateway. This topic primarily covers publishing outbound asynchronous messages from a PeopleSoft system to a third-party application using the delivered File Output connector. For information on outbound synchronous messages, see "Using Outbound Synchronous Messages" on page D-23.

To send a message, you must properly configure various internal structures and processes. The following descriptions are generally release-generic. Detailed differences between releases 8.1 and 8.4 are discussed in other topics.

- Message

  A Message is a container for the data that goes into the XML. It contains basic structural information, such as records and fields. The Message must be in an Active status to send the XML file

- Message Channel

  The Message Channel is a mechanism for structuring records into logical groupings. Each Message can belong to only one Message Channel. The Message Channel must be in an Active (Run) status for the Message to be delivered.

In release 8.1, the Message Channel also provides preliminary routing instructions; you can specify which Message Nodes handle the message. Each Message Channel can route messages to multiple Message Nodes

- Message Node

  Message node functionality changed from 8.1 to 8.4:

  In release 8.1, the primary function of the Message Node is to specify which Gateway receives the messages.

  In release 8.4, much of the "intelligence" that was built into the Message Channel moved to the Message Node. This provides additional flexibility over release 8.1. You can specify which messages the Message Node can handle. In addition, the Gateway Connector is bound to the Message Node. Each Message Node can route messages to only one Connector.

- Integration Gateway

  The Integration Gateway is a program that runs on the PeopleSoft Web Server. It is the physical hub between PeopleSoft and the third-party system.

- Target Connector/Handler

  Connectors are Java programs that run under the control of the Integration Gateway and control the final output destination of the XML file. PeopleSoft release 8.4 comes with several connectors including `HTTP`, `FTP`, `SMTP`, `JMS`, `POP3`, and a `Simple File` connector that places the file in a directory on the Web Server. This section discusses the Simple File connector.

- PeopleCode

  PeopleCode is the programming tool provided with PeopleTools that enables you to create complex application functionality. A message can only be initiated using specific PeopleCode instructions. This code is typically triggered by an application event, such as creating a new database entry through an online panel or through a batch job.

Most of the examples in this section use the `LOCATION_SYNC` message, which is a PeopleSoft Enterprise Integration Point (EIP) and is supplied with most PeopleSoft applications. If `LOCATION_SYNC` is not part of your package, you may use any supplied message.

Configuring Integration Broker in PeopleSoft 8.4

You can configure PeopleSoft 8.4 to send an asynchronous outbound message to the File Output connector.

To configure application messaging in PeopleSoft 8.4:

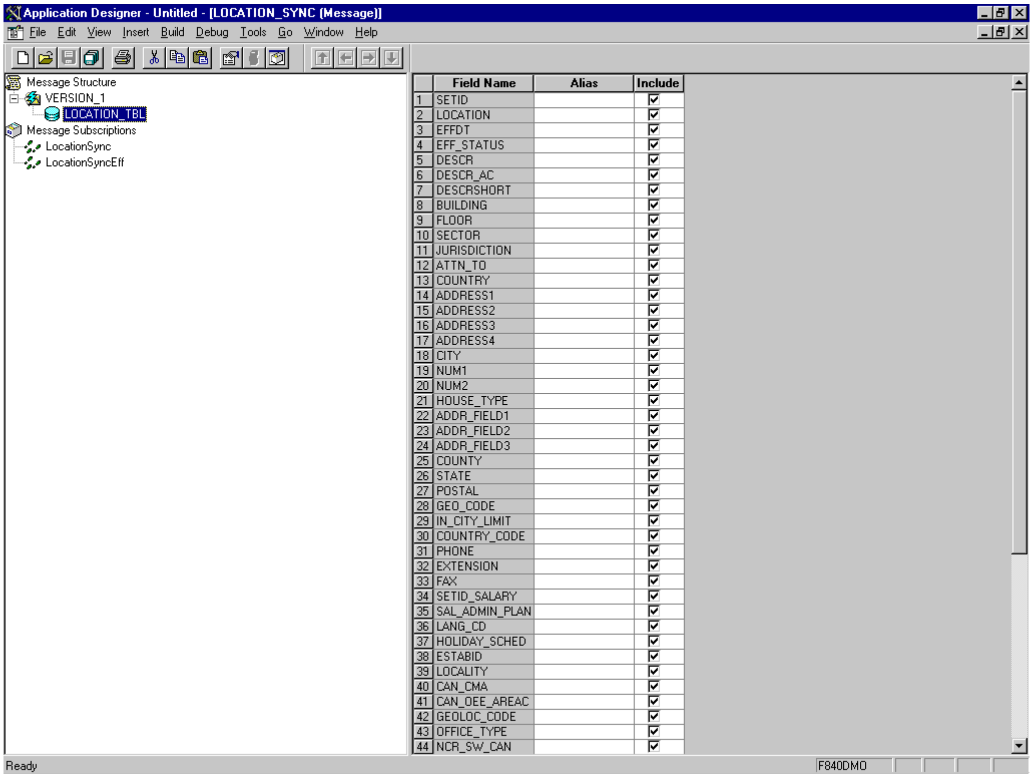
1. Ensure that the message is active and is routed to the proper Message Channel.
2. Configure the Message Channel.
3. Configure the IntegrationGateway.properties file to communicate with your PeopleSoft 8.4 application.
4. Configure the Integration Gateway and File Output connector.
5. Create and configure a new Gateway node.

   These tasks are described in detail in the following procedures.

Ensuring the Message Is Active and Is Routed Correctly

To ensure that the message is active and is routed to the proper Message Channel:

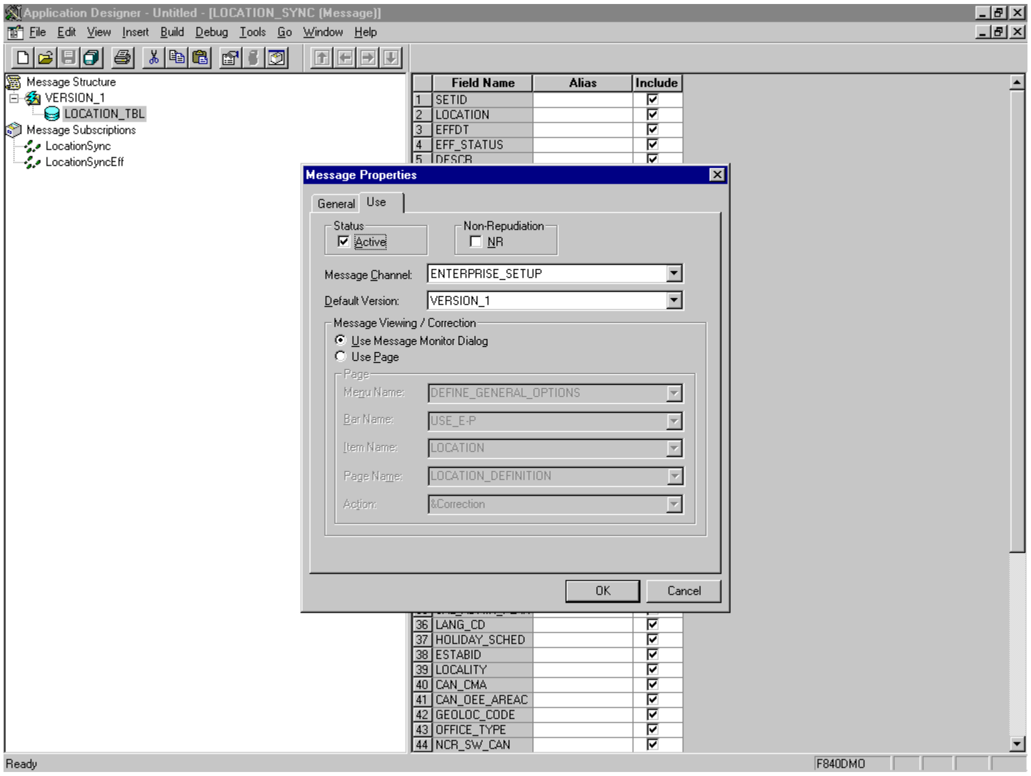
1. Open Application Designer.
2. On the **File** menu, point to **Open**, click **Message**, and then open the **LOCATION_SYNC** message.


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Window Help
Message Structure
VERSION_1
LOCATION_TBL
Message Subscriptions
LocationSync
LocationSyncEff
Ready
F840DMO

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
6	DESCR_AC		☑
7	DESCRSHORT		☑
8	BUILDING		☑
9	FLOOR		☑
10	SECTOR		☑
11	JURISDICTION		☑
12	ATTN_TO		☑
13	COUNTRY		☑
14	ADDRESS1		☑
15	ADDRESS2		☑
16	ADDRESS3		☑
17	ADDRESS4		☑
18	CITY		☑
19	NUM1		☑
20	NUM2		☑
21	HOUSE_TYPE		☑
22	ADDR_FIELD1		☑
23	ADDR_FIELD2		☑
24	ADDR_FIELD3		☑
25	COUNTY		☑
26	STATE		☑
27	POSTAL		☑
28	GEO_CODE		☑
29	IN_CITY_LIMIT		☑
30	COUNTRY_CODE		☑
31	PHONE		☑
32	EXTENSION		☑
33	FAX		☑
34	SETID_SALARY		☑
35	SAL_ADMIN_PLAN		☑
36	LANG_CD		☑
37	HOLIDAY_SCHED		☑
38	ESTABID		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

3. To view the fields that are included in the Message, highlight **LOCATION_TBL**.
4. Right-click **LOCATION_TBL** and select **Properties**.

   The Message Properties dialog box is displayed.


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Window Help
Message Structure
VERSION_1
LOCATION_TBL
Message Subscriptions
LocationSync
LocationSyncEff
Ready
F840DMO

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
36	LANG_CD		☑
37	HOLIDAY_SCHED		☑
38	ESTABID		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

Message Properties

General | Use

Status: ☑ Active

Non-Repudiation: □ NR

Message Channel: ENTERPRISE_SETUP

Default Version: VERSION_1

Message Viewing / Correction

● Use Message Monitor Dialog

○ Use Page

Page

Menu Name: DEFINE_GENERAL_OPTIONS

Bar Name: USE_E-P

Item Name: LOCATION

Page Name: LOCATION_DEFINITION

Action: &Correction

OK | Cancel

5. Select the **Use** tab.
6. Ensure the **Active** check box is selected.

   The message is routed to the Message Channel, `ENTERPRISE_SETUP`, and the default message version is VERSION_1 (messages can have multiple versions).
7. Click **OK** and then save the message.

   You have finished ensuring that the message is active and is routed correctly.

Configuring the IntegrationGateway.properties File

To configure the `IntegrationGateway.properties` file:

1. Open the `IntegrationGateway.properties` file using the editor of your choice.
2. Find the section of the file that specifies the JOLT connect string setting for the default application server. This is usually near line 75, and looks similar to the following:

   ```
   ## JOLT connect string setting for optional Default Application Server. Do NOT
   specify a NODENAME.
   #
   # Example:
   #ig.isc.serverURL=//MYSERVER:9000
   #ig.isc.userid=MYUSERID
   #ig.isc.password=MYPASSWORD
   #ig.isc.toolsRel=8.40
   ```
3. Uncomment (or copy and uncomment) the four lines that specify the connection.
4. Enter the appropriate information.

   In the following example, the tools release is 8.40.09:

```
ig.isc.serverURL=//isdsrv14:9000
ig.isc.userid=VP1
ig.isc.password=VP1
ig.isc.toolsRel=8.40.09
```

The PeopleSoft tools release must be precise to the last decimal.

> **Note:** With release 8.42, the password must be stored in an encrypted format. PeopleSoft provides a script called `PSCipher.bat` (`PSCipher.sh` on UNIX) to accomplish encryption. Typically, this script is located in the path of the `IntegrationGateway.properties` file. Follow the instructions supplied by PeopleSoft to run this script.

You have finished configuring the `IntegrationGateway.properties` file.

Configuring the Integration Gateway and the File Output Connector

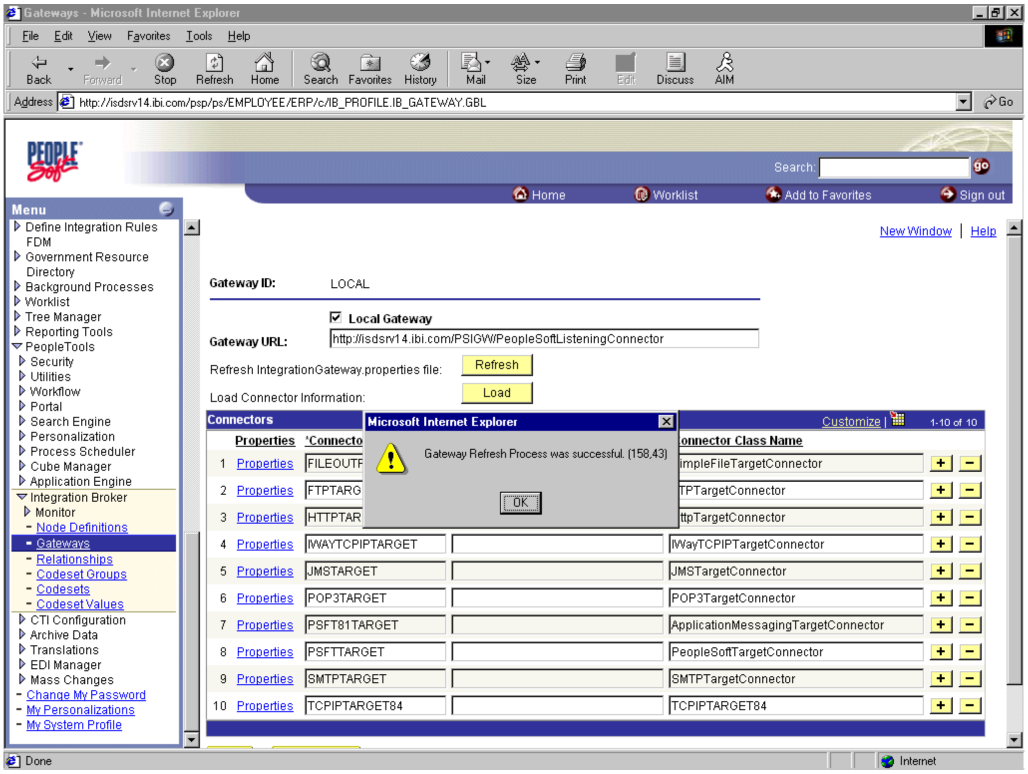
To configure the Integration Gateway and the File Output Connector:

1. In a Web browser, open your PeopleSoft 8.4 application in 4-tier mode.
2. In the Menu pane, expand **PeopleTools**, **Integration Broker**, and then click **Gateways**.
3. Open the LOCAL Gateway ID and enter the following Gateway URL:

   ```
   hostname/PSIGW/PeopleSoftListeningConnector
   ```

   Where `hostname` is the name of your PeopleSoft Web Server.
4. Click **Refresh**.

   A message appears stating the outcome of the refresh process.



Gateway ID: LOCAL

☑ Local Gateway

Gateway URL: http://isdsrv14.ibi.com/PSIGW/PeopleSoftListeningConnector

Refresh IntegrationGateway.properties file: Refresh

Load Connector Information: Load

Microsoft Internet Explorer

Gateway Refresh Process was successful. (158,43)

OK

Connectors — Customize | 1-10 of 10

	Properties	*Connecto		onnector Class Name		
1	Properties	FILEOUTF		impleFileTargetConnector	+	−
2	Properties	FTPTARG		TPTargetConnector	+	−
3	Properties	HTTPTAR		ttpTargetConnector	+	−
4	Properties	IWAYTCPIPTARGET		IWayTCPIPTargetConnector	+	−
5	Properties	JMSTARGET		JMSTargetConnector	+	−
6	Properties	POP3TARGET		POP3TargetConnector	+	−
7	Properties	PSFT81TARGET		ApplicationMessagingTargetConnector	+	−
8	Properties	PSFTTARGET		PeopleSoftTargetConnector	+	−
9	Properties	SMTPTARGET		SMTPTargetConnector	+	−
10	Properties	TCPIPTARGET84		TCPIPTARGET84	+	−

5. Click **OK** and scroll down to click **Save**.

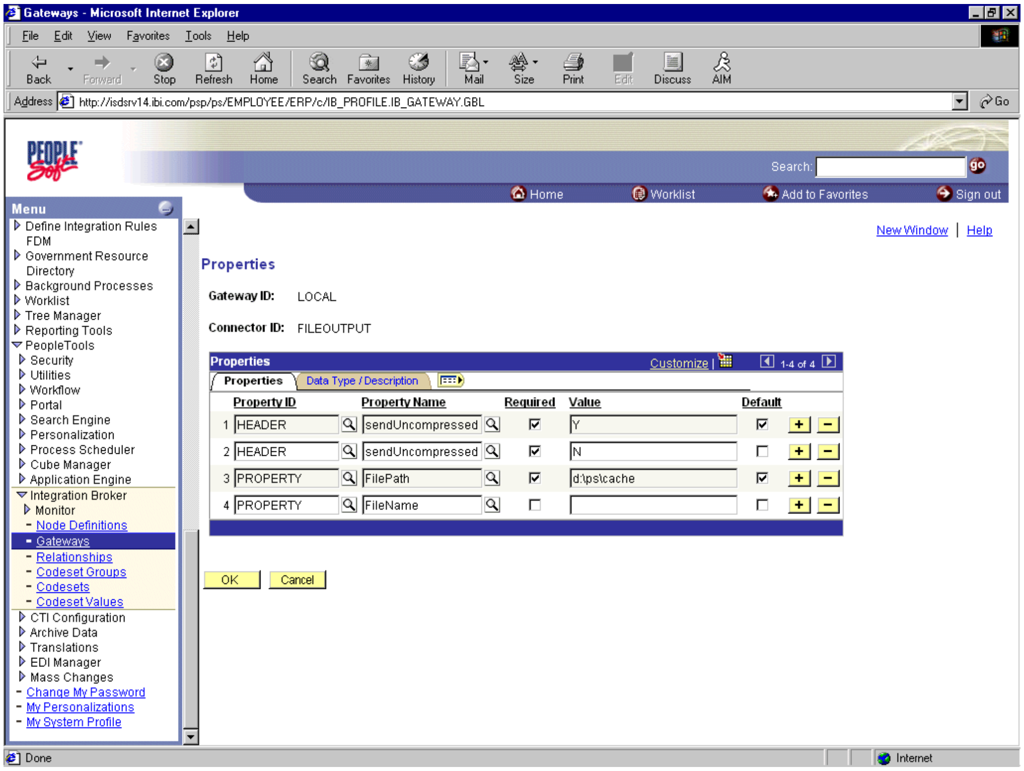
   You must click Save before continuing.

6. Click the **Properties** link for the FILEOUTPUT Connector ID.

   The Properties window for the FILEOUTPUT Connector is displayed.

7. Accept or overwrite the default values.

   In the following figure, the FilePath PROPERTY from the `c:\temp default` was changed to `d:\ps\cache`.



Properties

Gateway ID: LOCAL

Connector ID: FILEOUTPUT

Properties

	Property ID	Property Name	Required	Value	Default
1	HEADER	sendUncompressed	☑	Y	☑
2	HEADER	sendUncompressed	☑	N	□
3	PROPERTY	FilePath	☑	d:\ps\cache	☑
4	PROPERTY	FileName	□		□

8. To return to the Gateway window, click **OK**.

9. Scroll down and click **Save**.

   You have finished configuring the Integration Gateway and the File Output Connector.

Creating and Configuring a New Gateway Node

To create and configure a new Gateway Node:

1. In the Menu pane, expand **PeopleTools**, **Integration Broker**, and then, click **Node Definitions**.

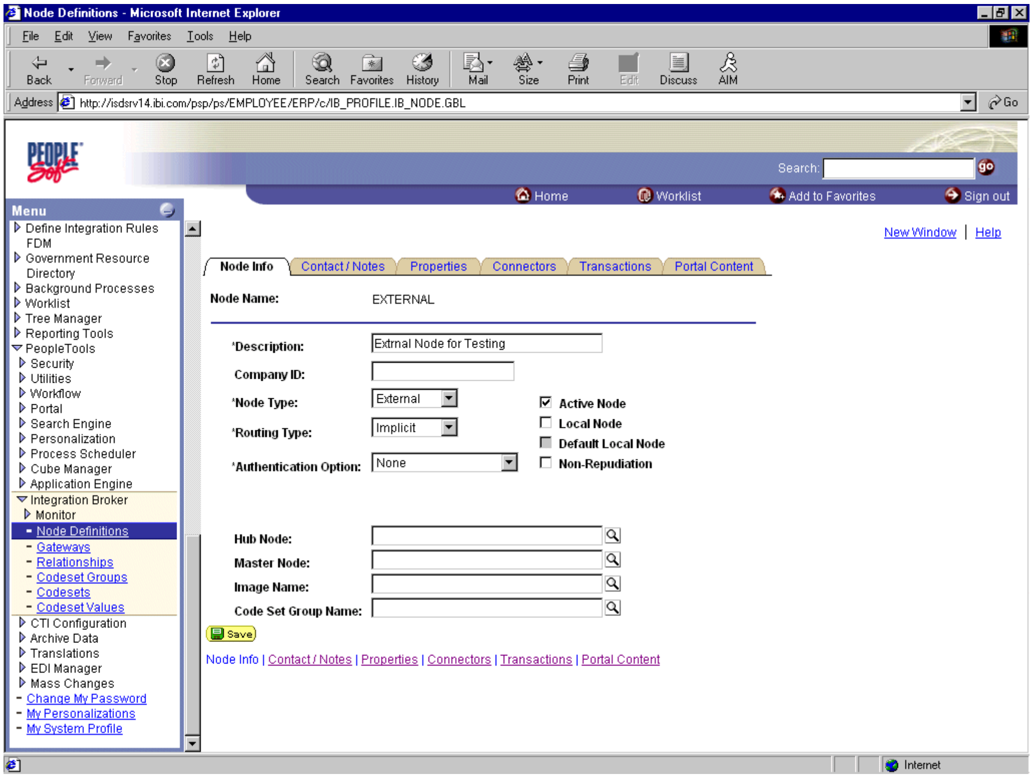
2. Select the **Add a New Value** tab.

3. In the **Node Name** field, enter a node name.

   It is recommended that you name your first (trial) message node EXTERNAL. After successfully configuring and sending messages using this node, you can create additional message nodes with names appropriate for your application.

4. Click **Add**.

The Node Info tab becomes available.



Node Definitions - Microsoft Internet Explorer

Address: http://isdsrv14.ibi.com/psp/ps/EMPLOYEE/ERP/c/IB_PROFILE.IB_NODE.GBL

Node Info | Contact / Notes | Properties | Connectors | Transactions | Portal Content

Node Name: EXTERNAL

*Description: Extrnal Node for Testing

Company ID:

*Node Type: External

*Routing Type: Implicit

*Authentication Option: None

☑ Active Node
☐ Local Node
☐ Default Local Node
☐ Non-Repudiation

Hub Node:

Master Node:

Image Name:

Code Set Group Name:

Perform the following steps:

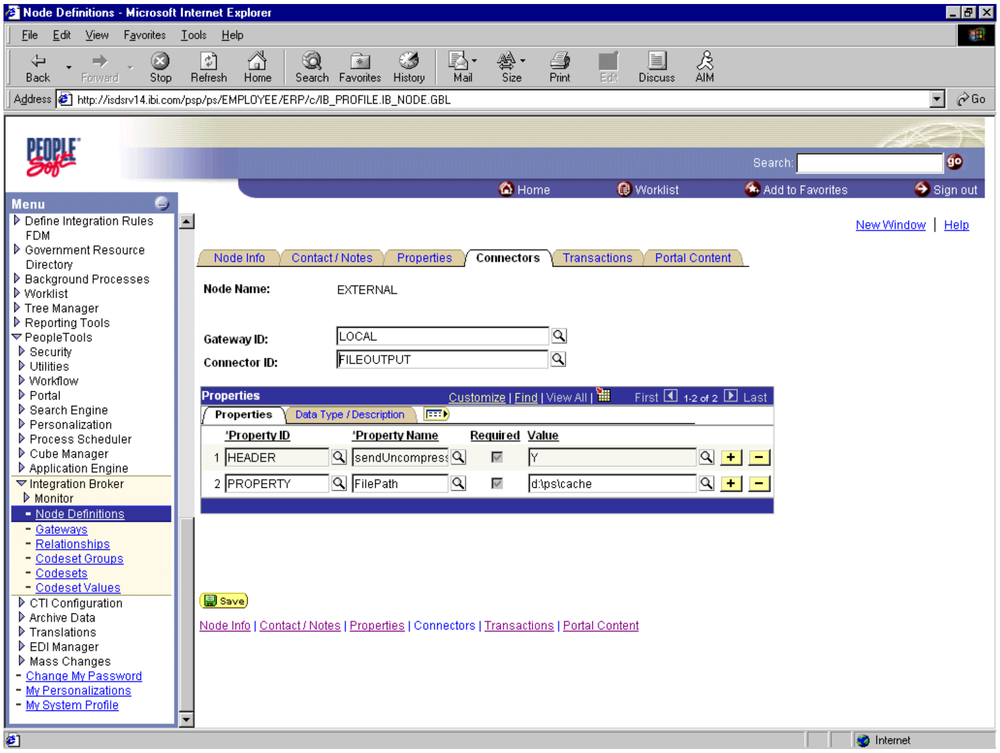
a. In the **Description** field, enter an appropriate description.

b. From the **Node Type** list, select **EXTERNAL**.

c. From the **Routing Type** list, select **Implicit**.

5. Select the **Connectors** tab.

The Connectors tab becomes available.



Node Name: EXTERNAL

Gateway ID: LOCAL

Connector ID: FILEOUTPUT

Properties

	*Property ID	*Property Name	Required	Value
1	HEADER	sendUncompress	☑	Y
2	PROPERTY	FilePath	☑	d:\ps\cache

Perform the following steps:

a. Specify **LOCAL** for the **Gateway ID**.

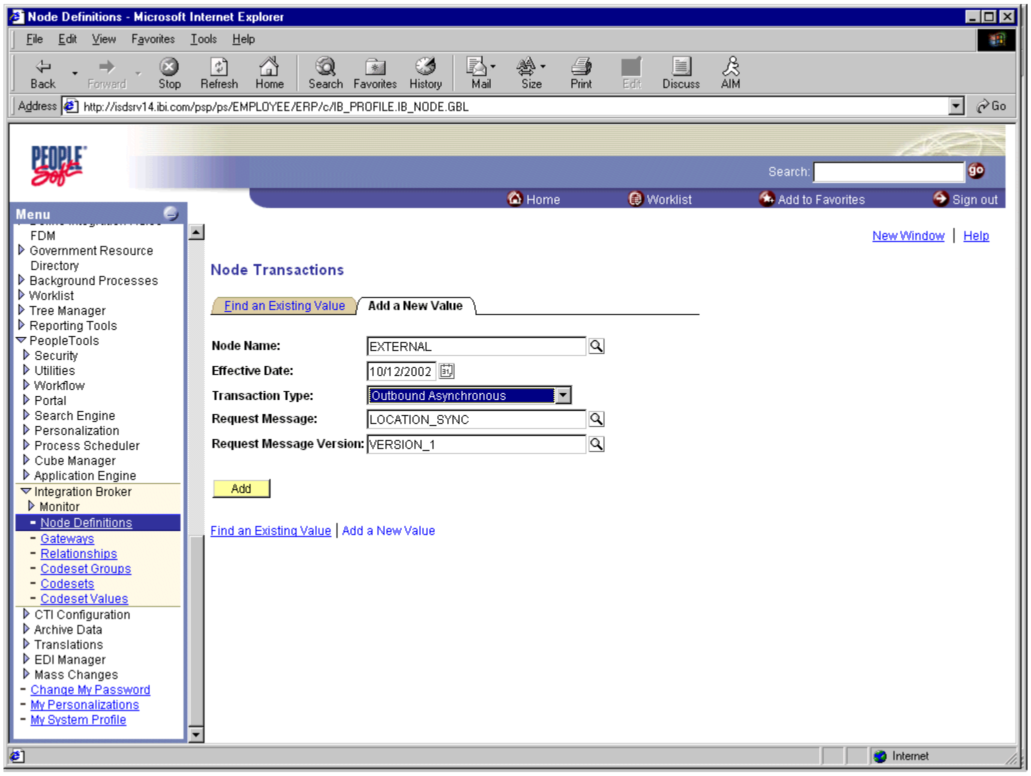
b. Specify **FILEOUTPUT** for the **Connector ID**.

c. Accept or overwrite the default Gateway property values.

6. Click **Save**.

7. To specify the transactions to route messages to your node, select the **Transactions** tab and click **Add Transaction**.

   The Node Transactions pane is displayed.



Node Transactions

Node Name:	EXTERNAL
Effective Date:	10/12/2002
Transaction Type:	Outbound Asynchronous
Request Message:	LOCATION_SYNC
Request Message Version:	VERSION_1

Perform the following steps:

a. From the **Transaction Type** list, select **Outbound Asynchronous**.

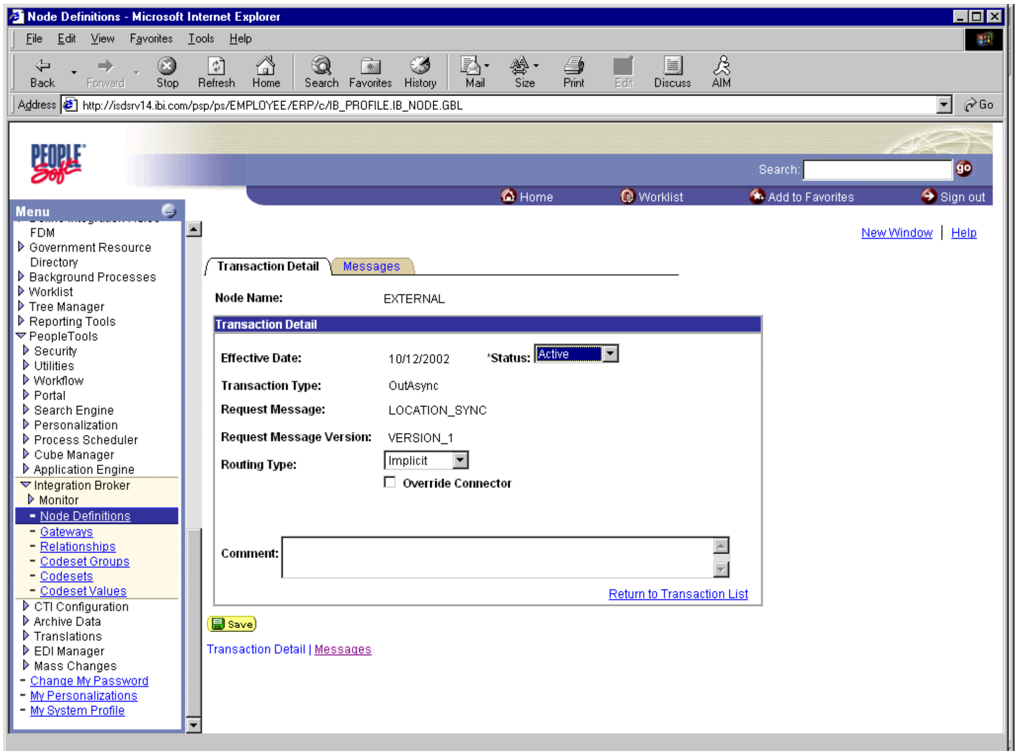
b. In the **Request Message** field, specify **LOCATION_SYNC**.

c. In the **Request Message Version** field, specify **VERSION_1**.

In the **Request Message Version** field, specify **VERSION_1**.

8. Click **Add**.

The Transaction Detail pane is displayed.



Node Name: EXTERNAL

Transaction Detail

Effective Date: 10/12/2002 *Status: Active

Transaction Type: OutAsync

Request Message: LOCATION_SYNC

Request Message Version: VERSION_1

Routing Type: Implicit

□ Override Connector

Comment:

Return to Transaction List

Perform the following steps:

a. Verify that the **Routing Type** is **Implicit**.

b. Click **Save**, then click the **Return to Transaction List** link.

c. To ensure that your data entry is not lost, click **Save** again.

You have finished creating and configuring the new Gateway Node.

See "Viewing the PeopleCode for a Message" on page D-25 for more information.

Configuring Application Messaging in PeopleSoft Release 8.1

You can configure PeopleSoft 8.1 to send an asynchronous outbound message to the Simple File Handler.

To configure application messaging in PeopleSoft 8.1:

1. Create and configure a new Message Node.
2. Ensure the message is active and is routed to the proper Message Channel.
3. Configure the Message Channel.
4. Configure the Simple File Handler in the Gateway.

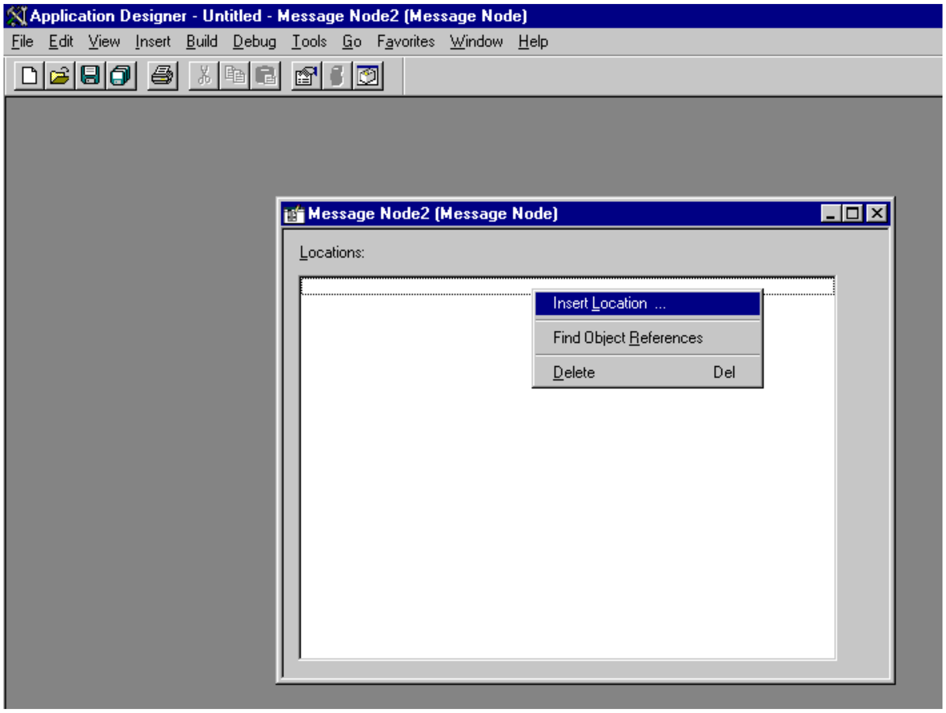
   These tasks are described in detail in the following procedures.

Creating and Configuring a New Message Node

To create and configure a new message node:

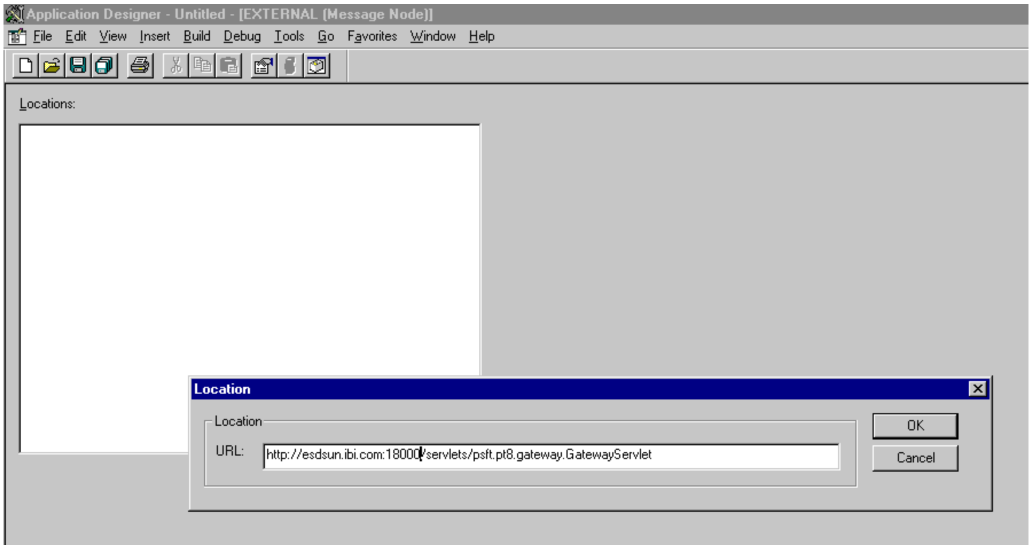
1. Select **New** from the File menu and click **Message Node**.

   A Message Node window is displayed.


Application Designer - Untitled - Message Node2 (Message Node)
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Node2 (Message Node)
Locations:
Insert Location ...
Find Object References
Delete Del

2. Right-click anywhere inside the white space and select **Insert Location**.

   The Location URL box is displayed.


Application Designer - Untitled - [EXTERNAL (Message Node)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Locations:
Location
Location
OK
Cancel

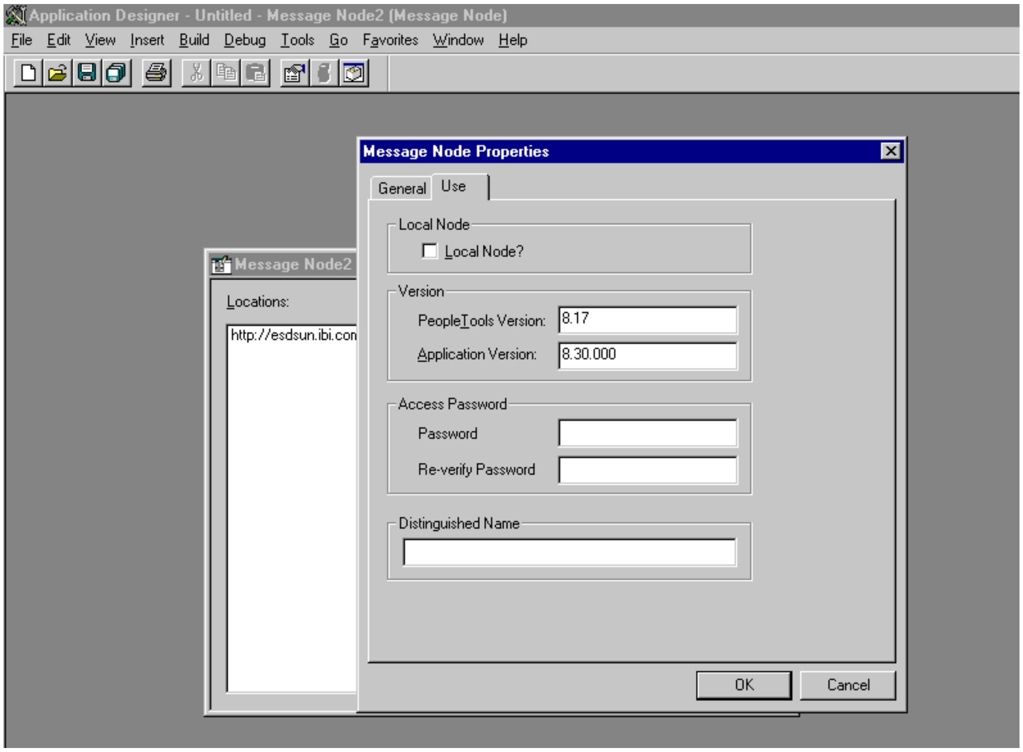
URL: http://esdsun.ibi.com:18000/servlets/psft.pt8.gateway.GatewayServlet

3. Enter the following URL for the PeopleSoft Application Gateway (handler directory):

   hostname:*port*/servlets/psft.pt8.gateway.GatewayServlet

Where `hostname` is the host name of your PeopleSoft Web server and `port` is the socket on which the server is listening. The characters you enter after `hostname` are case-sensitive.

4. Click **OK**.
5. Display the Message Node Properties dialog box.


Application Designer - Untitled - Message Node2 (Message Node)
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Node2
Locations:
http://esdsun.ibi.con
Message Node Properties
General Use
OK
Cancel

Local Node
□ Local Node?

Version
PeopleTools Version: 8.17
Application Version: 8.30.000

Access Password
Password
Re-verify Password

Distinguished Name

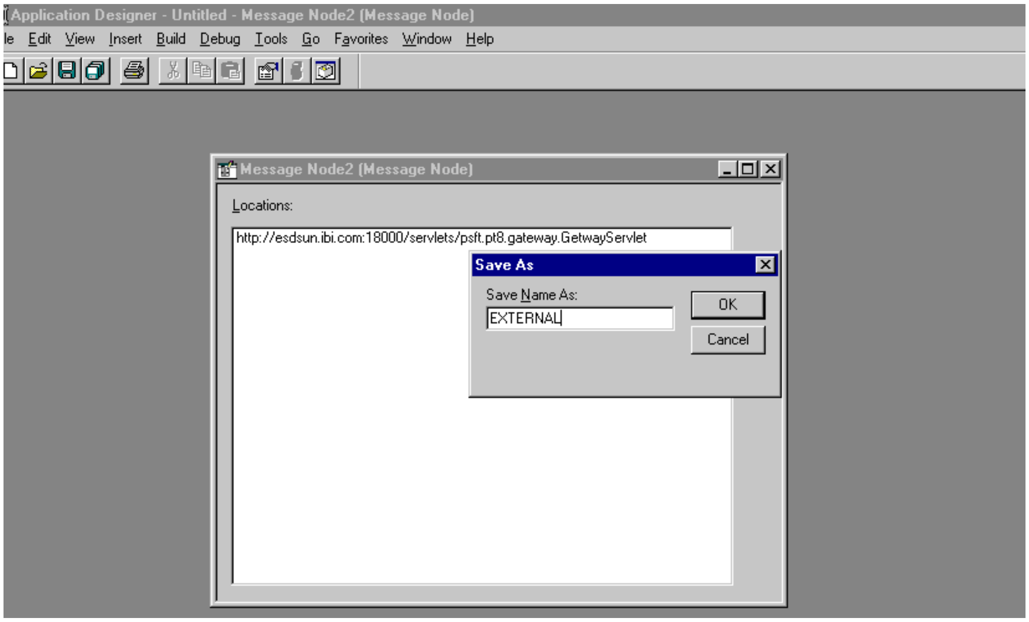
Perform the following steps:

a. Select the **Use** tab.

b. In the text boxes, enter the PeopleTools and Application Version numbers.

c. Click **OK**.

6. Display the Save As dialog box.


Application Designer - Untitled - Message Node2 (Message Node)
le Edit View Insert Build Debug Tools Go Favorites Window Help
Message Node2 (Message Node)
Save As
OK
Cancel

Locations:
http://esdsun.ibi.com:18000/servlets/psft.pt8.gateway.GetwayServlet

Save Name As: EXTERNAL

7. To save the Message Node, click **OK**.

It is recommended that you name your first (trial) message node EXTERNAL. After successfully configuring and sending messages using this node, you can create additional message nodes with names appropriate for your application.

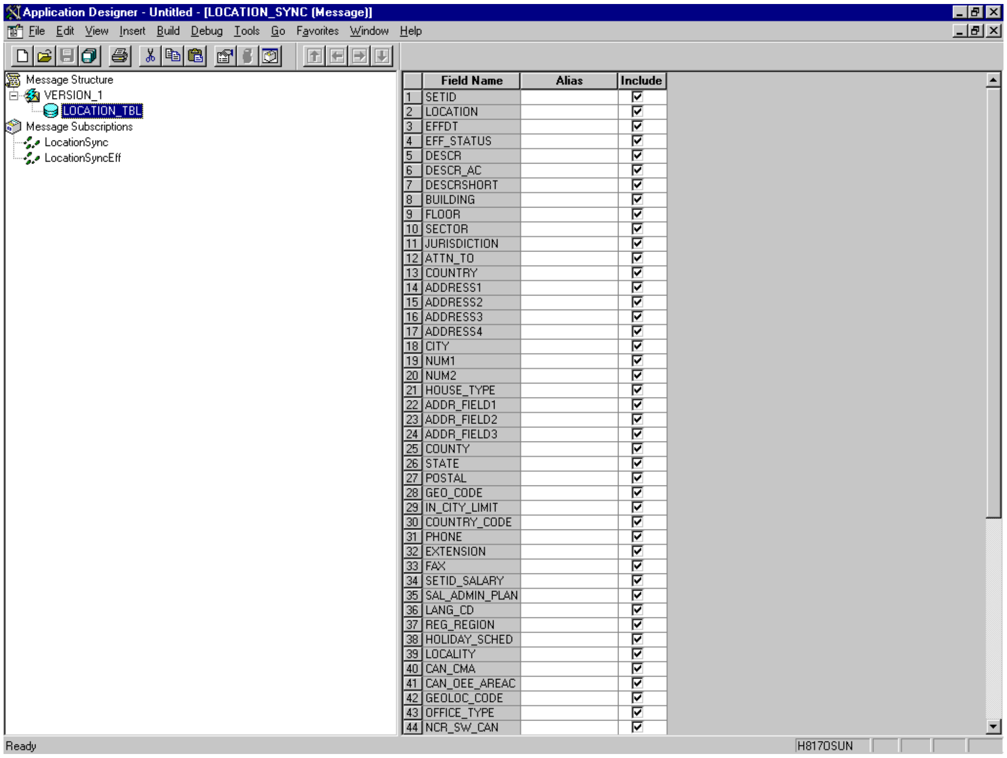
If you intend to migrate this message node to a different PeopleSoft environment (for example, from Test to QA), you can create a PeopleSoft project and insert the Message Node into the project.

You have finished creating and configuring the message node.

Ensuring the Message Is Active and Is Routed Correctly

To ensure that the message is active and is routed to the proper message channel:

1. Open Application Designer.
2. On the **File** menu, point to **Open**, click **Message**, and open the **LOCATION_SYNC** message.

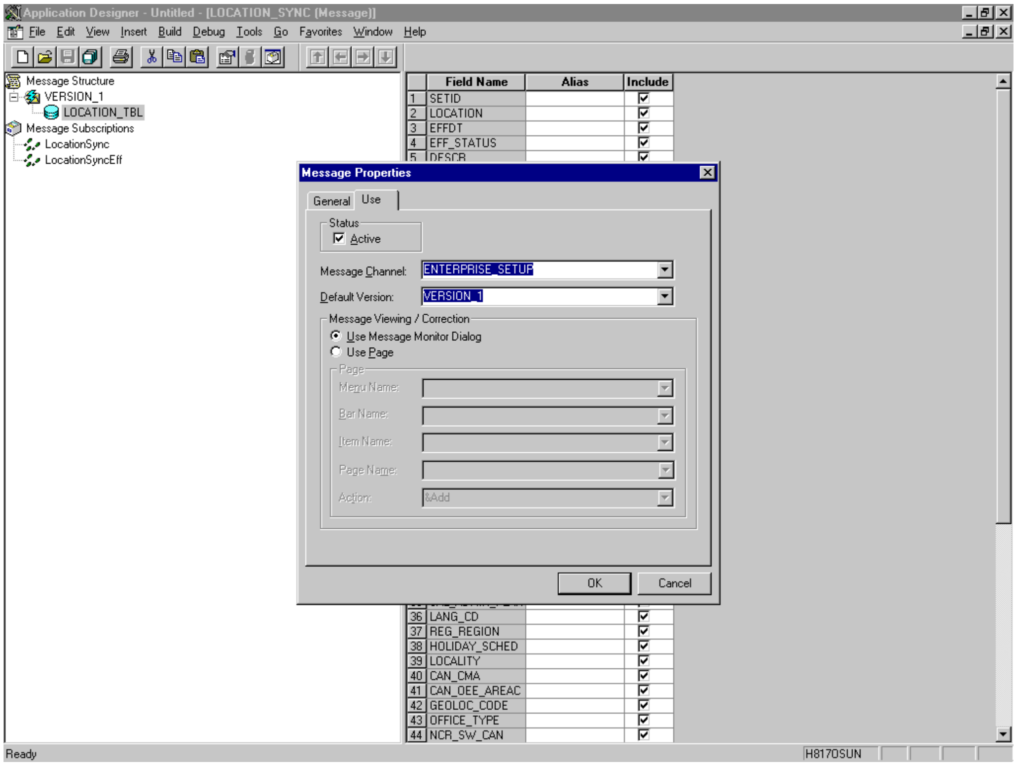


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Structure
VERSION_1
LOCATION_TBL
Message Subscriptions
LocationSync
LocationSyncEff
Ready
H817OSUN

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
6	DESCR_AC		☑
7	DESCRSHORT		☑
8	BUILDING		☑
9	FLOOR		☑
10	SECTOR		☑
11	JURISDICTION		☑
12	ATTN_TO		☑
13	COUNTRY		☑
14	ADDRESS1		☑
15	ADDRESS2		☑
16	ADDRESS3		☑
17	ADDRESS4		☑
18	CITY		☑
19	NUM1		☑
20	NUM2		☑
21	HOUSE_TYPE		☑
22	ADDR_FIELD1		☑
23	ADDR_FIELD2		☑
24	ADDR_FIELD3		☑
25	COUNTY		☑
26	STATE		☑
27	POSTAL		☑
28	GEO_CODE		☑
29	IN_CITY_LIMIT		☑
30	COUNTRY_CODE		☑
31	PHONE		☑
32	EXTENSION		☑
33	FAX		☑
34	SETID_SALARY		☑
35	SAL_ADMIN_PLAN		☑
36	LANG_CD		☑
37	REG_REGION		☑
38	HOLIDAY_SCHED		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

3. To view the fields that are included in the message, highlight **LOCATION_TBL**.
4. Right-click **LOCATION_TBL** and select **Properties**.

The Message Properties dialog box is displayed.


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Structure
VERSION_1
LOCATION_TBL
Message Subscriptions
LocationSync
LocationSyncEff
General
Use
OK
Cancel
Ready
H8170SUN

Message Properties

Status

☑ Active

Message Channel: ENTERPRISE_SETUP

Default Version: VERSION_1

Message Viewing / Correction

● Use Message Monitor Dialog

○ Use Page

Page

Menu Name:

Bar Name:

Item Name:

Page Name:

Action: &Add

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
36	LANG_CD		☑
37	REG_REGION		☑
38	HOLIDAY_SCHED		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

Perform the following steps:

a. Select the **Use** tab.

b. Ensure the **Status** check box is selected, which indicates that the message is active.

c. From the Message Channel list, select **ENTERPRISE_SETUP**.

d. From the Default Version list, select **VERSION_1** (messages can have multiple versions).

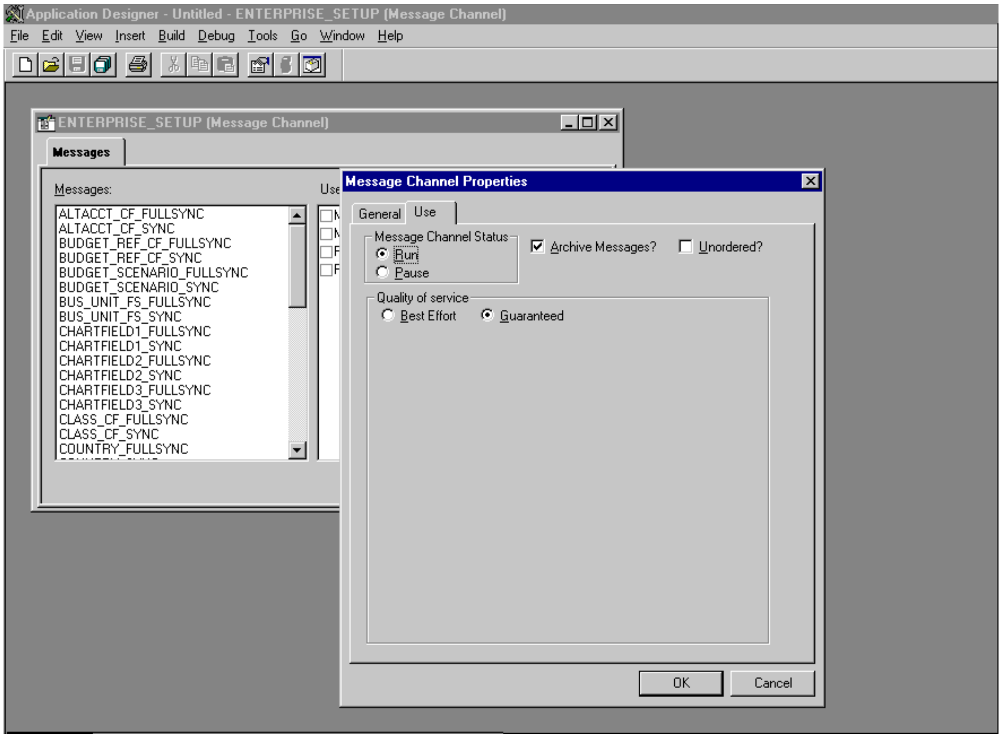
5. Click **OK**.
6. Save the message.

Configuring the Message Channel

To configure the message channel:

1. From the **File** menu, choose **Open** and click **Message Channel**.
2. To open the **ENTERPRISE_SETUP Message Channel**, select **ENTERPRISE_SETUP**.
3. Right-click **ENTERPRISE_SETUP** and select **Properties**.

   The Message Channel Properties dialog box is displayed.


Application Designer - Untitled - ENTERPRISE_SETUP (Message Channel)
File Edit View Insert Build Debug Tools Go Window Help
ENTERPRISE_SETUP (Message Channel)
Messages
Message Channel Properties
General
Use
OK
Cancel

Messages:

ALTACCT_CF_FULLSYNC
ALTACCT_CF_SYNC
BUDGET_REF_CF_FULLSYNC
BUDGET_REF_CF_SYNC
BUDGET_SCENARIO_FULLSYNC
BUDGET_SCENARIO_SYNC
BUS_UNIT_FS_FULLSYNC
BUS_UNIT_FS_SYNC
CHARTFIELD1_FULLSYNC
CHARTFIELD1_SYNC
CHARTFIELD2_FULLSYNC
CHARTFIELD2_SYNC
CHARTFIELD3_FULLSYNC
CHARTFIELD3_SYNC
CLASS_CF_FULLSYNC
CLASS_CF_SYNC
COUNTRY_FULLSYNC

Message Channel Status ● Run ○ Pause

☑ Archive Messages? □ Unordered?

Quality of service ○ Best Effort ● Guaranteed

Perform the following steps:

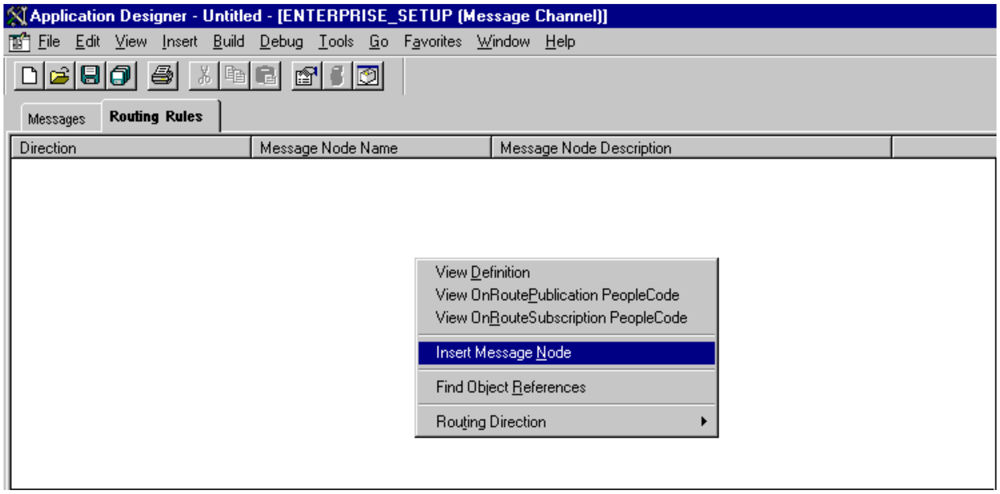
a. Select the **Use** tab.

b. Ensure that the Message Channel status is set to **Run**.

c. Click **OK**.

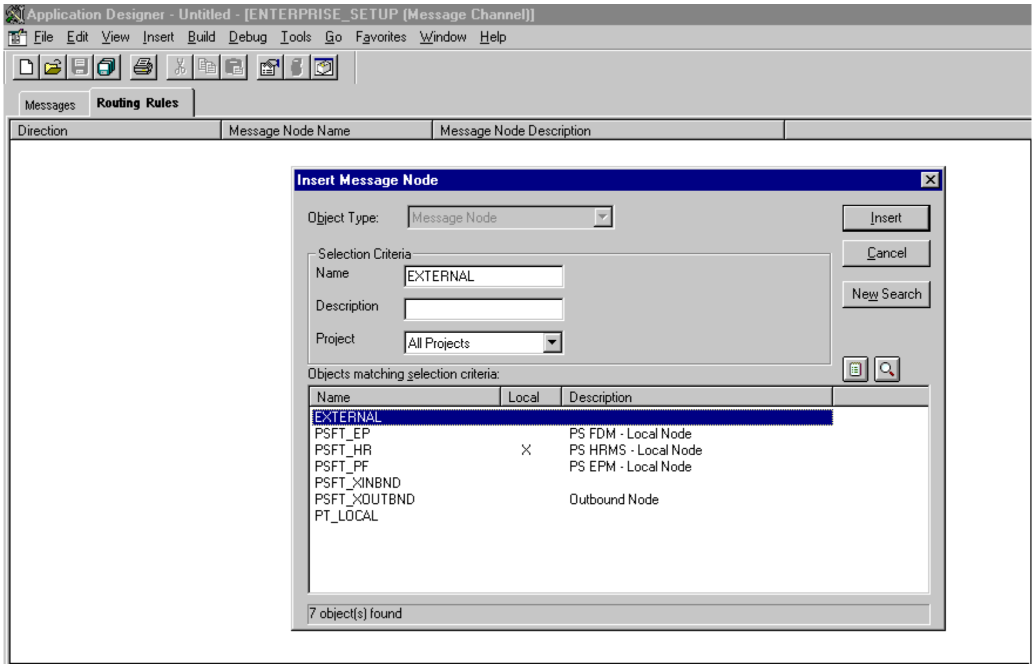
4. From the left pane, select the **Routing Rules** tab.

   The pane is blank.


Application Designer - Untitled - [ENTERPRISE_SETUP (Message Channel)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Messages
Routing Rules
Direction
Message Node Name
Message Node Description
View Definition
View OnRoutePublication PeopleCode
View OnRouteSubscription PeopleCode
Insert Message Node
Find Object References
Routing Direction

5. Right-click the pane and select **Insert Message Node**.

   The Insert Message Node dialog box is displayed.


Application Designer - Untitled - [ENTERPRISE_SETUP (Message Channel)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Messages
Routing Rules
Direction
Message Node Name
Message Node Description
Insert Message Node
Insert
Cancel
New Search

Object Type: Message Node

Selection Criteria

Name: EXTERNAL

Description:

Project: All Projects

Objects matching selection criteria:

Name	Local	Description
EXTERNAL		
PSFT_EP		PS FDM - Local Node
PSFT_HR	X	PS HRMS - Local Node
PSFT_PF		PS EPM - Local Node
PSFT_XINBND		
PSFT_XOUTBND		Outbound Node
PT_LOCAL		

7 object(s) found

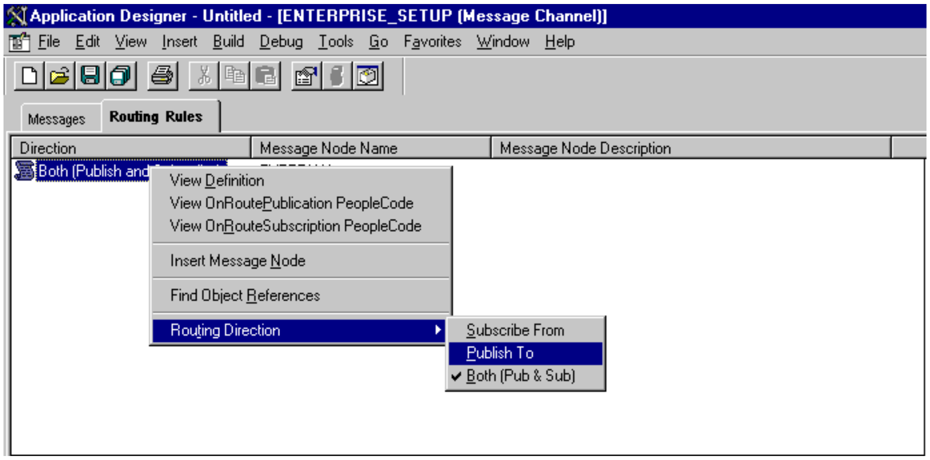
Perform the following steps:

a. Select the message node that you created in "Creating and Configuring a New Message Node" on page D-11, for example, EXTERNAL.

b. Click **Insert**.

6. Click **Cancel**.

Information appears on the Routing Rules tab.


Application Designer - Untitled - [ENTERPRISE_SETUP (Message Channel)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Messages
Routing Rules
Direction
Message Node Name
Message Node Description
Both (Publish and
View Definition
View OnRoutePublication PeopleCode
View OnRouteSubscription PeopleCode
Insert Message Node
Find Object References
Routing Direction
Subscribe From
Publish To
✓ Both (Pub & Sub)

Perform the following steps:

a. Right-click the message node and point to **Routing Direction**.

b. From the **Routing Direction** menus, select **Publish To**.

7. Save the Message Channel, and if you require it, place it in your project.

You have finished configuring the Message Channel.

Configuring the Simple File Handler in the Gateway

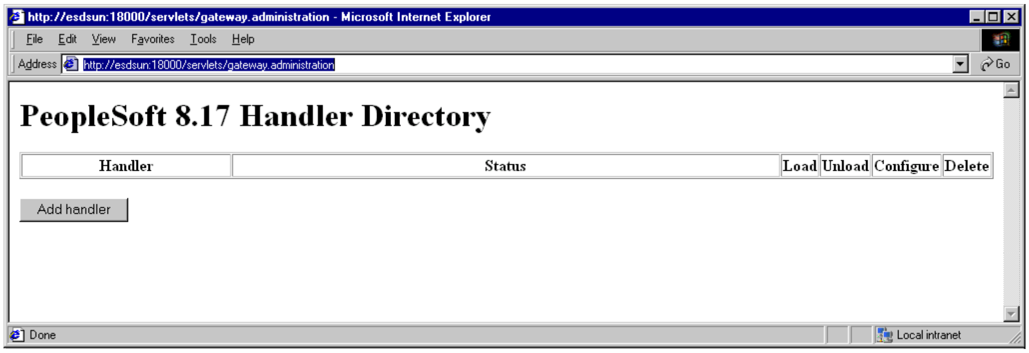
To configure the Simple File Handler in the Gateway:

1. In a Web browser, launch the PeopleSoft 8.1 configuration servlet interface (also known as the server gateway) by entering the following URL:

   ```
   hostname:port/servlets/gateway.administration
   ```

   Where `hostname` is the name of the application server that hosts PeopleSoft and `port` is the port number on which the application server is listening.

   The Handler Directory window is displayed.



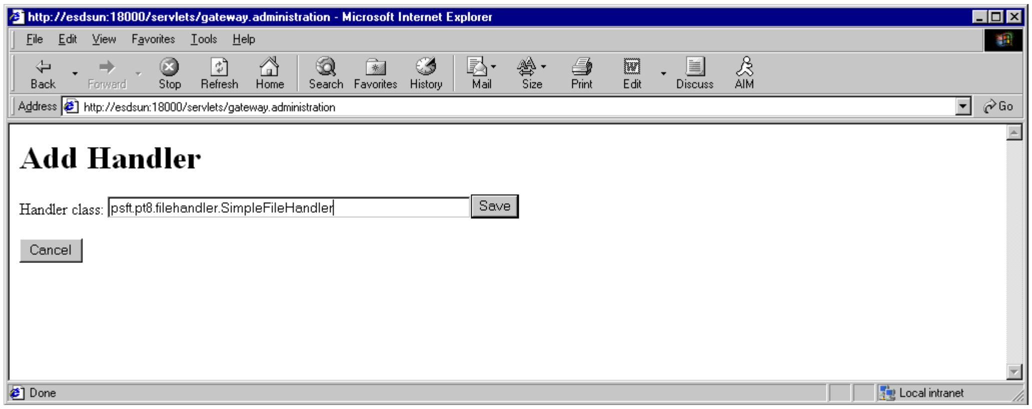
http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://esdsun:18000/servlets/gateway.administration
Go
Add handler
Done
Local intranet

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete

2. Click **Add Handler**.

   The Add Handler window is displayed.



http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Save
Cancel
Done
Local intranet

Add Handler

Handler class: psft.pt8.filehandler.SimpleFileHandler

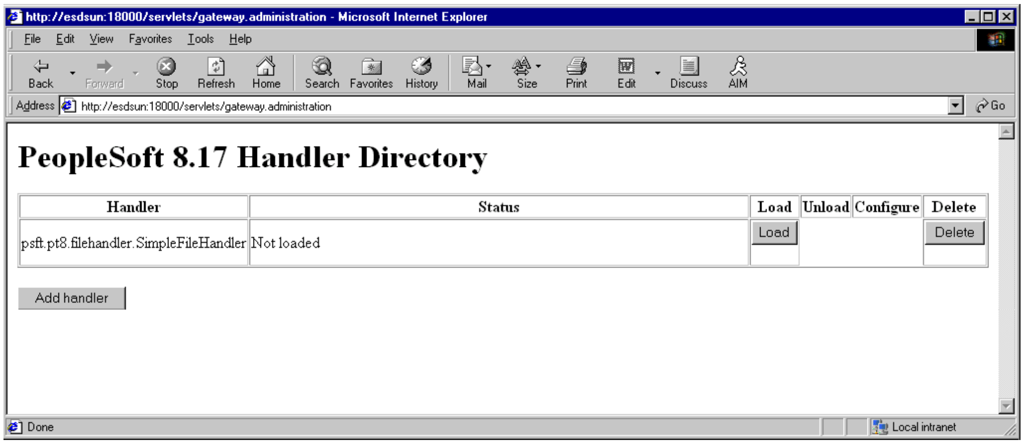
3. Enter the full name of the Simple File Handler class, `psft.pt8.filehandler.SimpleFileHandler`.

   Note: The name is case-sensitive.

   ---

4. Click **Save**.

   The Handler Directory window is displayed.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete
psft.pt8.filehandler.SimpleFileHandler	Not loaded	Load			Delete

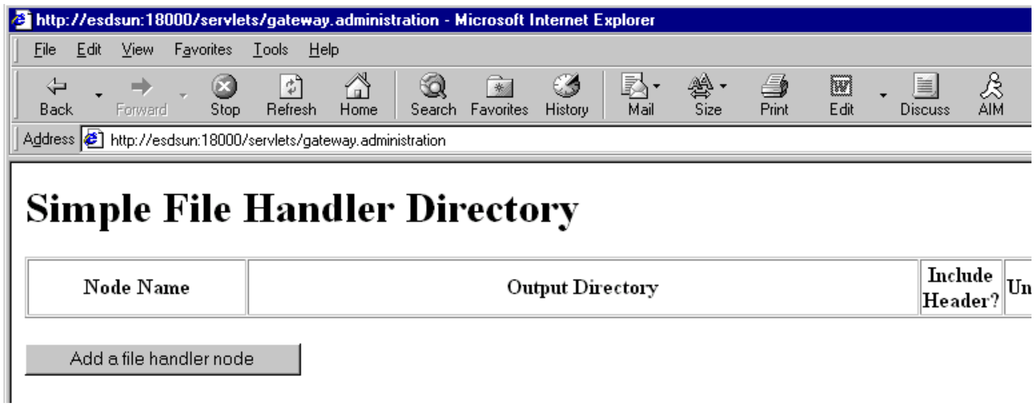
Add handler

5. To load the handler, click **Load**.

   After the handler loads, "Loaded successfully" appears in the Status column.

6. Click **Configure**.

   The Simple File Handler Directory window is displayed.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration

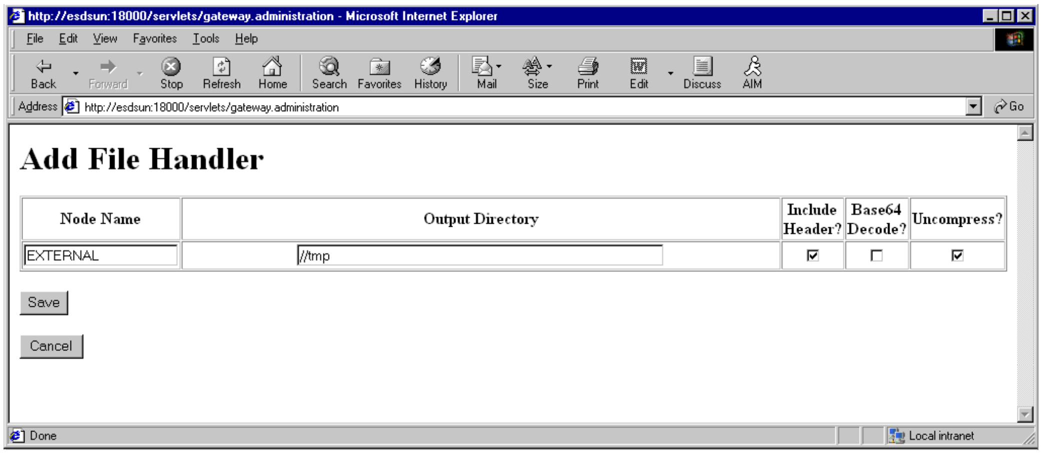
Simple File Handler Directory

Node Name	Output Directory	Include Header?	Un

Add a file handler node

7. Click **Add a file handler node**.

   The Add File Handler window is displayed.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

Add File Handler

Node Name	Output Directory	Include Header?	Base64 Decode?	Uncompress?
EXTERNAL	//tmp	☑	☐	☑

Save

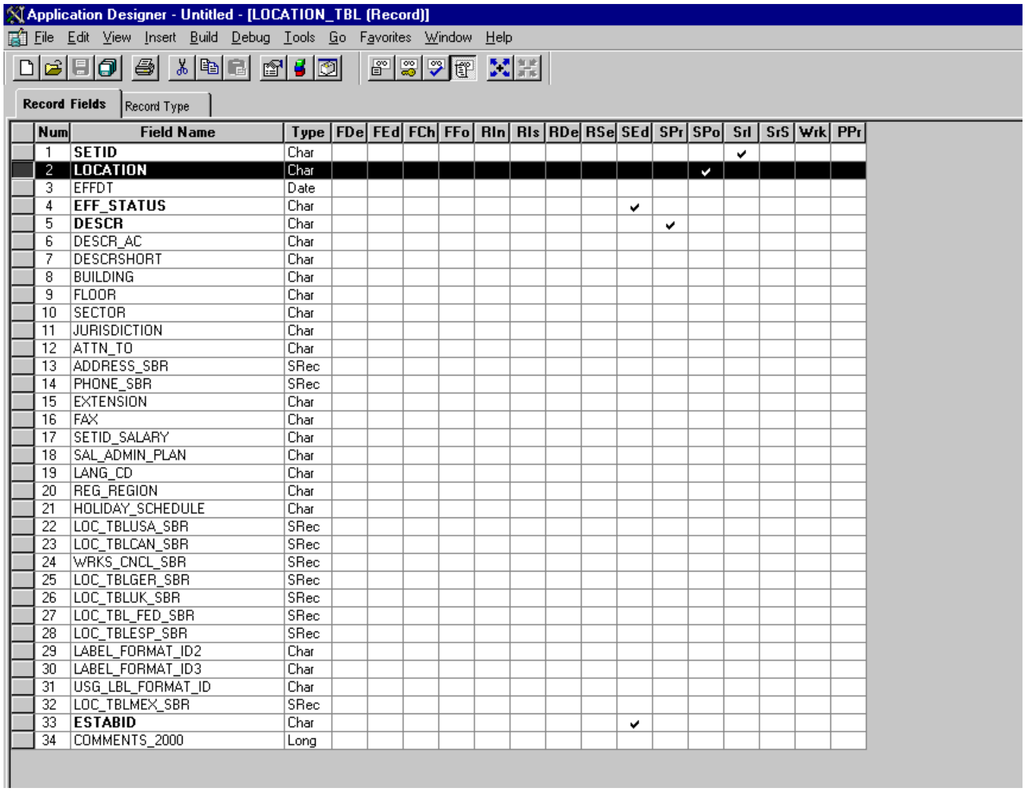
Cancel

8. In the Node Name field, enter the name of the message node that you created in "Creating and Configuring a New Message Node" on page D-11, for example, EXTERNAL.

Viewing the PeopleCode for a Message

Messages are initiated by the PeopleCode that is attached to a record. Usually, this record is one of the records associated with the message itself.

1. Open Application Designer.


Application Designer - Untitled - [LOCATION_TBL (Record)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Record Fields
Record Type

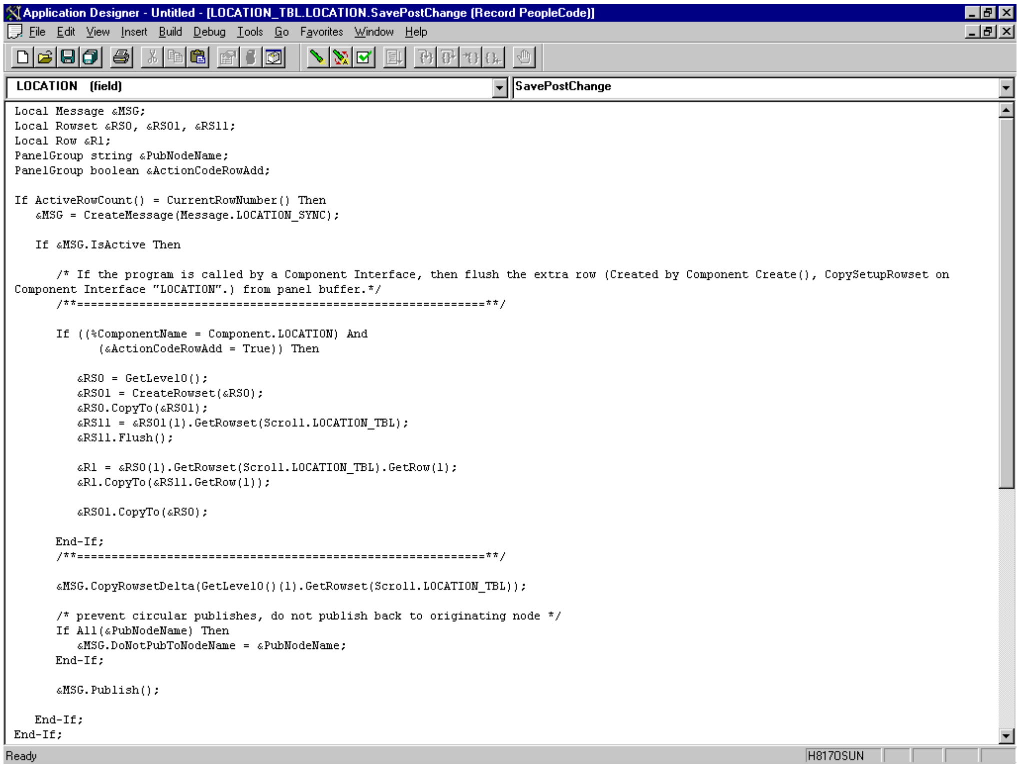
Num	Field Name	Type	FDe	FEd	FCh	FFo	RIn	RIs	RDe	RSe	SEd	SPr	SPo	SrI	SrS	Wrk	PPr
1	**SETID**	Char												✓			
2	**LOCATION**	Char											✓				
3	EFFDT	Date															
4	**EFF_STATUS**	Char									✓						
5	**DESCR**	Char										✓					
6	DESCR_AC	Char															
7	DESCRSHORT	Char															
8	BUILDING	Char															
9	FLOOR	Char															
10	SECTOR	Char															
11	JURISDICTION	Char															
12	ATTN_TO	Char															
13	ADDRESS_SBR	SRec															
14	PHONE_SBR	SRec															
15	EXTENSION	Char															
16	FAX	Char															
17	SETID_SALARY	Char															
18	SAL_ADMIN_PLAN	Char															
19	LANG_CD	Char															
20	REG_REGION	Char															
21	HOLIDAY_SCHEDULE	Char															
22	LOC_TBLUSA_SBR	SRec															
23	LOC_TBLCAN_SBR	SRec															
24	WRKS_CNCL_SBR	SRec															
25	LOC_TBLGER_SBR	SRec															
26	LOC_TBLUK_SBR	SRec															
27	LOC_TBL_FED_SBR	SRec															
28	LOC_TBLESP_SBR	SRec															
29	LABEL_FORMAT_ID2	Char															
30	LABEL_FORMAT_ID3	Char															
31	USG_LBL_FORMAT_ID	Char															
32	LOC_TBLMEX_SBR	SRec															
33	**ESTABID**	Char									✓						
34	COMMENTS_2000	Long															

2. On the **Record Fields** tab, select the **LOCATION_TBL** record.

3. Select the PeopleCode display option.

4. Select the **Save Post Change (SPo)** box for the **LOCATION** field.

The following window displays the PeopleCode that initiates a LOCATION_SYNC message.


Application Designer - Untitled - [LOCATION_TBL.LOCATION.SavePostChange (Record PeopleCode)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Ready
H817OSUN

LOCATION (field) SavePostChange

```
Local Message &MSG;
Local Rowset &RS0, &RS01, &RS11;
Local Row &R1;
PanelGroup string &PubNodeName;
PanelGroup boolean &ActionCodeRowAdd;

If ActiveRowCount() = CurrentRowNumber() Then
   &MSG = CreateMessage(Message.LOCATION_SYNC);

   If &MSG.IsActive Then

      /* If the program is called by a Component Interface, then flush the extra row (Created by Component Create(), CopySetupRowset on
Component Interface "LOCATION".) from panel buffer.*/
      /**===========================================================**/

      If ((%ComponentName = Component.LOCATION) And
            (&ActionCodeRowAdd = True)) Then

         &RS0 = GetLevel0();
         &RS01 = CreateRowset(&RS0);
         &RS0.CopyTo(&RS01);
         &RS11 = &RS01(1).GetRowset(Scroll.LOCATION_TBL);
         &RS11.Flush();

         &R1 = &RS0(1).GetRowset(Scroll.LOCATION_TBL).GetRow(1);
         &R1.CopyTo(&RS11.GetRow(1));

         &RS01.CopyTo(&RS0);

      End-If;
      /**===========================================================**/

      &MSG.CopyRowsetDelta(GetLevel0()(1).GetRowset(Scroll.LOCATION_TBL));

      /* prevent circular publishes, do not publish back to originating node */
      If All(&PubNodeName) Then
         &MSG.DoNotPubToNodeName = &PubNodeName;
      End-If;

      &MSG.Publish();

   End-If;
End-If;
```

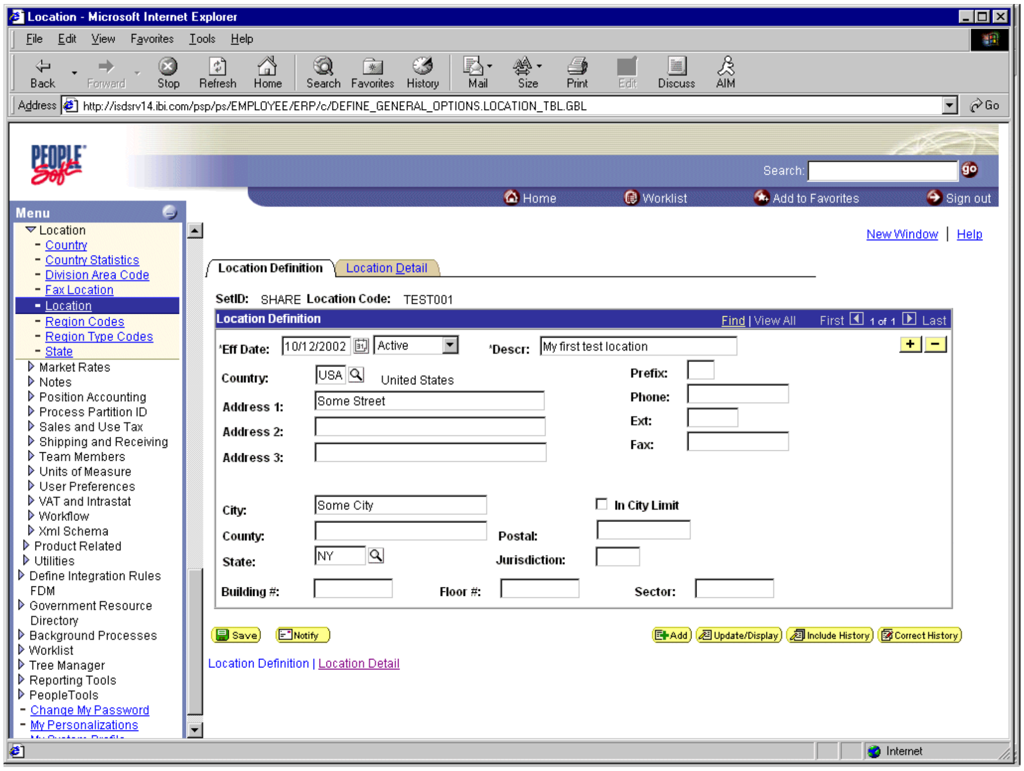
For more information about PeopleCode, consult your PeopleSoft Online Library.

You have finished viewing the PeopleCode for a message. You can now test Integration Broker (in PeopleSoft 8.4) or Application Messaging (in PeopleSoft 8.1).

Testing the Integration Broker

To test the Integration Broker by generating a message, you can navigate to the Location Transaction window and add, update, or delete a location entry in your application. Depending on your application, the way you navigate varies.

The following example illustrates a Financials 8.4 application where a new location with a SetID of SHARE and a Location Code of TEST001 was added.



Address: http://isdsrv14.ibi.com/psp/ps/EMPLOYEE/ERP/c/DEFINE_GENERAL_OPTIONS.LOCATION_TBL.GBL

SetID: SHARE Location Code: TEST001

Location Definition

*Eff Date: 10/12/2002 Active *Descr: My first test location

Country: USA United States

Address 1: Some Street

Address 2:

Address 3:

Prefix:

Phone:

Ext:

Fax:

City: Some City

□ In City Limit

County:

Postal:

State: NY

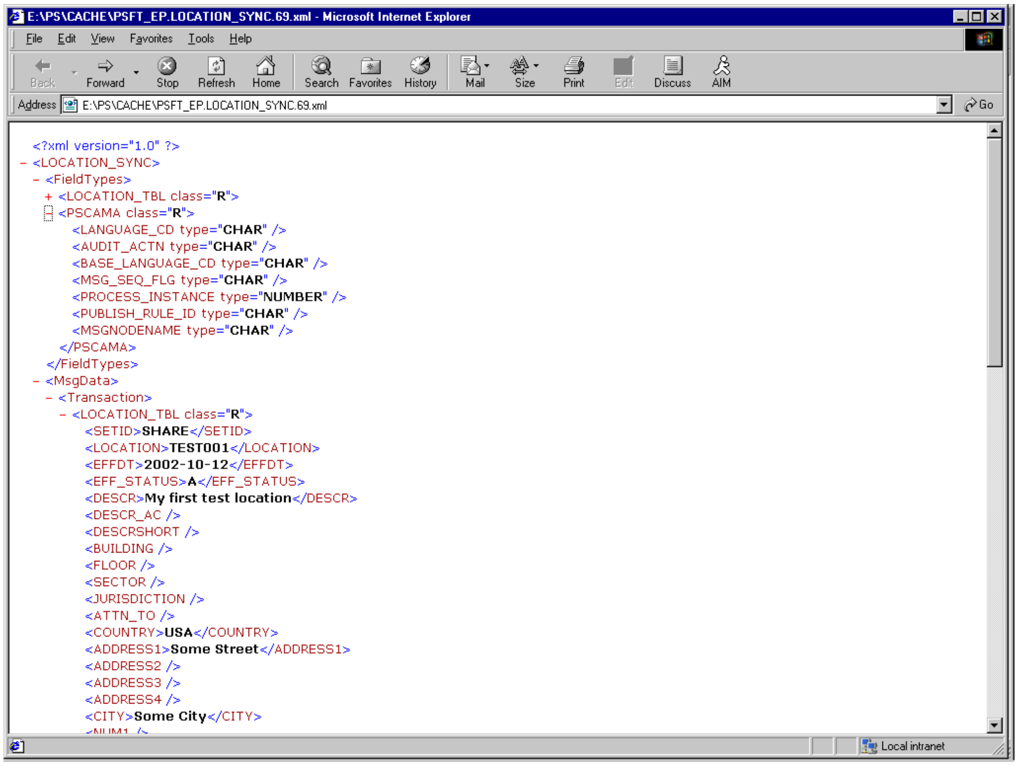
Jurisdiction:

Building #:

Floor #:

Sector:

The following figure shows a portion of the XML output.


E:\PS\CACHE\PSFT_EP.LOCATION_SYNC.69.xml - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address
Go
Local intranet

Address: E:\PS\CACHE\PSFT_EP.LOCATION_SYNC.69.xml

```
  <?xml version="1.0" ?>
- <LOCATION_SYNC>
  - <FieldTypes>
    + <LOCATION_TBL class="R">
    - <PSCAMA class="R">
        <LANGUAGE_CD type="CHAR" />
        <AUDIT_ACTN type="CHAR" />
        <BASE_LANGUAGE_CD type="CHAR" />
        <MSG_SEQ_FLG type="CHAR" />
        <PROCESS_INSTANCE type="NUMBER" />
        <PUBLISH_RULE_ID type="CHAR" />
        <MSGNODENAME type="CHAR" />
      </PSCAMA>
    </FieldTypes>
  - <MsgData>
    - <Transaction>
      - <LOCATION_TBL class="R">
          <SETID>SHARE</SETID>
          <LOCATION>TEST001</LOCATION>
          <EFFDT>2002-10-12</EFFDT>
          <EFF_STATUS>A</EFF_STATUS>
          <DESCR>My first test location</DESCR>
          <DESCR_AC />
          <DESCRSHORT />
          <BUILDING />
          <FLOOR />
          <SECTOR />
          <JURISDICTION />
          <ATTN_TO />
          <COUNTRY>USA</COUNTRY>
          <ADDRESS1>Some Street</ADDRESS1>
          <ADDRESS2 />
          <ADDRESS3 />
          <ADDRESS4 />
          <CITY>Some City</CITY>
```

Note: The name of the file is `PSFT_EP.LOCATION_SYNC.69.xml`, which is the concatenation of `PSFT_EP` (the local Publishing Node), the name of the message, and the number of the Publication ID.

If you cannot send a message successfully, PeopleSoft provides a set of tools for monitoring the progress of your messages. In release 8.1, you use a tool called the Application Messaging Monitor. In release 8.4, you use the Monitor Menu in the Integration Broker.

For a complete description on how to isolate and resolve problems with your messaging environment, consult your PeopleSoft Online Library. If you are still unable to send your XML file, the PeopleSoft Customer Connection can help solve your problem.

Using Outbound Synchronous Messages

Starting with PeopleTools 8.4, you can send outbound synchronous messages. From a high-level point of view, the primary difference between outbound synchronous and asynchronous is that with outbound asynchronous, the transaction is completed whether or not the message is actually sent or received.

For synchronous outbound messages:

- The transaction must wait for a response from the external system before continuing.
- The transaction must process the response message.
- The external system must ensure that the response message is correctly formatted.

OracleAS Adapter for PeopleSoft can work with PeopleSoft outbound synchronous messages. Outbound synchronous messages involve additional configuration steps, both within PeopleSoft and in Oracle Application Server. This topic briefly describes the configuration requirements within PeopleSoft.

Note: The instructions in this topic build upon the instructions for outbound asynchronous messages. It is strongly recommended that you familiarize yourself with outbound asynchronous messaging before attempting outbound synchronous. See "Configuring Integration Broker in PeopleSoft 8.4" on page D-2 for more information on outbound asynchronous messages.

Ensure that both outbound and inbound messages are created and active. PeopleSoft provides template examples called IB_INST_VER_SYNC_MSG and IB_INST_VER_RESP_MSG. F. See "Ensuring the Message Is Active and Is Routed Correctly" on page D-3 for information on examining these messages.

Configuring Outbound Synchronous Messages

You can use an existing node, or you can create a new node to configure outbound synchronous messages. See "Creating and Configuring a New Gateway Node" on page D-7 for information on creating and configuring a node. In either case, you must set up your outbound synchronous transaction.

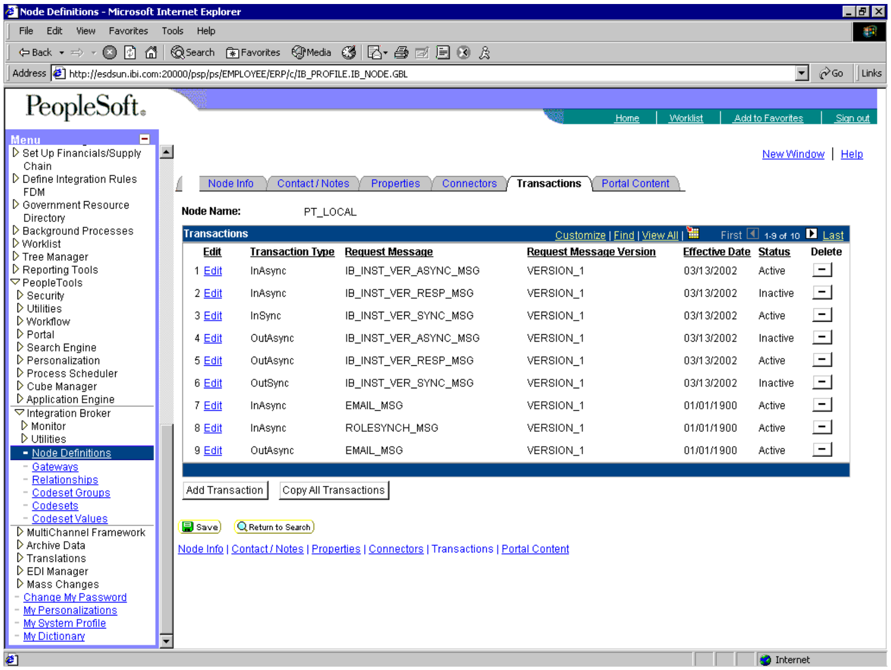
The following example uses a node and transaction delivered by PeopleSoft. However, this example is for illustrative purposes only and does not actually work as delivered without additional steps. As of Financials release 8.42, there are no preconfigured outbound synchronous transactions that you can use for testing purposes.

Configuring an Outbound Synchronous Message

To configure an outbound synchronous message:

1. Navigate to the **Node Definitions** page and open the **PT_LOCAL** node.
2. Click the **Transactions** tab.

   The Transactions pane is displayed.


Node Definitions - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://esdsun.ibi.com:20000/psp/ps/EMPLOYEE/ERP/c/IB_PROFILE.IB_NODE.GBL
Go Links
PeopleSoft.
Home | Worklist | Add to Favorites | Sign out
Menu
Set Up Financials/Supply Chain
Define Integration Rules FDM
Government Resource Directory
Background Processes
Worklist
Tree Manager
Reporting Tools
PeopleTools
Security
Utilities
Workflow
Portal
Search Engine
Personalization
Process Scheduler
Cube Manager
Application Engine
Integration Broker
Monitor
Utilities
Node Definitions
Gateways
Relationships
Codeset Groups
Codesets
Codeset Values
MultiChannel Framework
Archive Data
Translations
EDI Manager
Mass Changes
Change My Password
My Personalizations
My System Profile
My Dictionary
New Window | Help
Node Info | Contact / Notes | Properties | Connectors | Transactions | Portal Content
Customize | Find | View All | First 1-9 of 10 Last
Add Transaction
Copy All Transactions
Save
Return to Search
Node Info | Contact / Notes | Properties | Connectors | Transactions | Portal Content
Internet

Node Name: PT_LOCAL

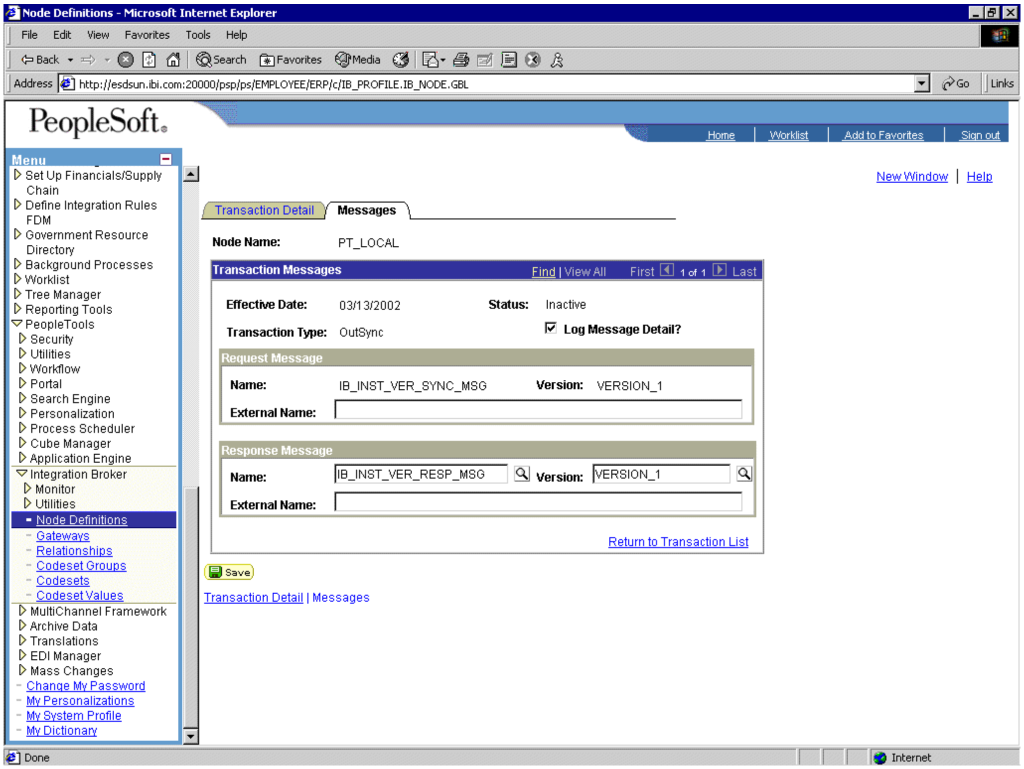
Transactions

	Edit	Transaction Type	Request Message	Request Message Version	Effective Date	Status	Delete
1	Edit	InAsync	IB_INST_VER_ASYNC_MSG	VERSION_1	03/13/2002	Active	−
2	Edit	InAsync	IB_INST_VER_RESP_MSG	VERSION_1	03/13/2002	Inactive	−
3	Edit	InSync	IB_INST_VER_SYNC_MSG	VERSION_1	03/13/2002	Active	−
4	Edit	OutAsync	IB_INST_VER_ASYNC_MSG	VERSION_1	03/13/2002	Inactive	−
5	Edit	OutAsync	IB_INST_VER_RESP_MSG	VERSION_1	03/13/2002	Active	−
6	Edit	OutSync	IB_INST_VER_SYNC_MSG	VERSION_1	03/13/2002	Inactive	−
7	Edit	InAsync	EMAIL_MSG	VERSION_1	01/01/1900	Active	−
8	Edit	InAsync	ROLESYNCH_MSG	VERSION_1	01/01/1900	Active	−
9	Edit	OutAsync	EMAIL_MSG	VERSION_1	01/01/1900	Active	−

   One outbound synchronous message, `IB_INST_VER_SYNC_MSG`, appears in the Transaction Type list.

3. Click the **Edit** link in the `IB_INST_VER_SYNC_MSG` row.

   The following pane is displayed.



Node Name: PT_LOCAL

Transaction Messages

Effective Date: 03/13/2002 Status: Inactive

Transaction Type: OutSync ☑ Log Message Detail?

Request Message

Name: IB_INST_VER_SYNC_MSG Version: VERSION_1

External Name:

Response Message

Name: IB_INST_VER_RESP_MSG Version: VERSION_1

External Name:

4. Click the **Messages** tab.

Both request and response messages appear. The target system must ensure that the response message follows the format of the request message. As the target system is your Oracle Application Server, you must transform the XML that is sent and returned from your final destination.

> **Note:** You must use the PeopleSoft-supplied HTTP target connector when you are working with synchronous outbound messages. You cannot use the TCPIP84TARGET connector for outbound synchronous messages.

Viewing the PeopleCode for a Synchronous Message

The sample PeopleCode in the following example is for a synchronous outbound message. It differs from asynchronous outbound in that it must handle a response message.

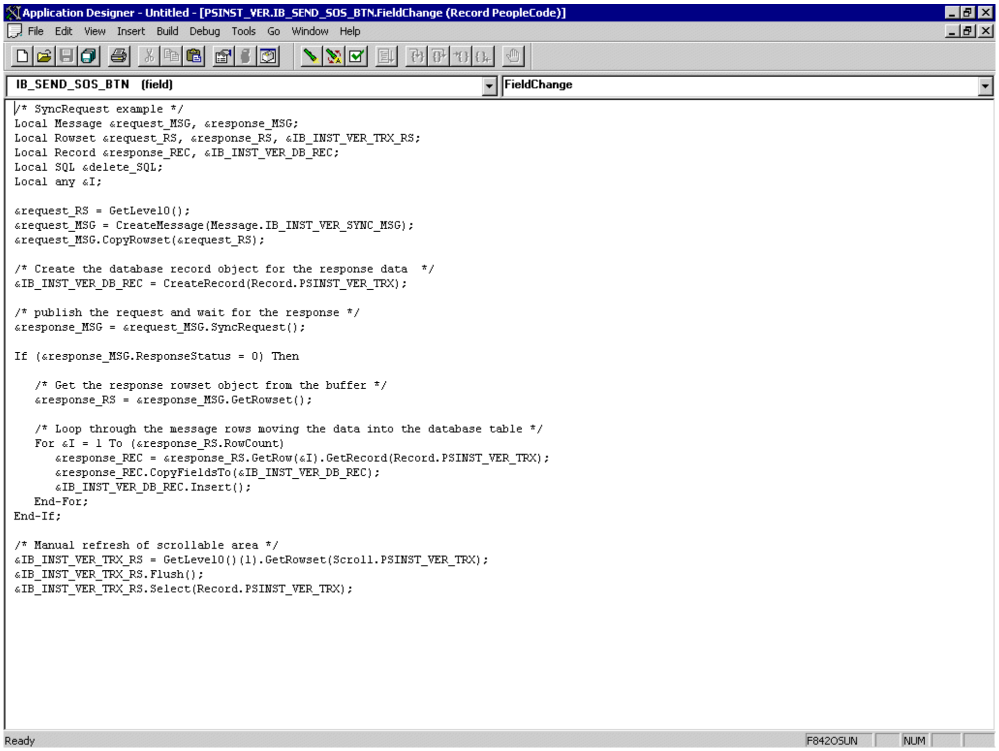
Viewing the PeopleCode for a Financials Synchronous Outbound Message

The following sample code is supplied with the Financials application and is associated with the two messages IB_INST_VER_SYNC_MSG and IB_INST_VER_RESP_MSG.

To view the code, perform the following steps:

1. From Application Designer, open the **PSINST_VER** record.
2. Select the **PeopleCode** display option.
3. Select the **Field Change (FCh)** box for the **IB_SEND_SOS_BTN** field.

The following window is displayed.



Application Designer - Untitled - [PSINST_VER.IB_SEND_SOS_BTN.FieldChange (Record PeopleCode)]
File Edit View Insert Build Debug Tools Go Window Help
IB_SEND_SOS_BTN (field)
FieldChange
Ready
F842OSUN
NUM

```
/* SyncRequest example */
Local Message &request_MSG, &response_MSG;
Local Rowset &request_RS, &response_RS, &IB_INST_VER_TRX_RS;
Local Record &response_REC, &IB_INST_VER_DB_REC;
Local SQL &delete_SQL;
Local any &I;

&request_RS = GetLevel0();
&request_MSG = CreateMessage(Message.IB_INST_VER_SYNC_MSG);
&request_MSG.CopyRowset(&request_RS);

/* Create the database record object for the response data  */
&IB_INST_VER_DB_REC = CreateRecord(Record.PSINST_VER_TRX);

/* publish the request and wait for the response */
&response_MSG = &request_MSG.SyncRequest();

If (&response_MSG.ResponseStatus = 0) Then

   /* Get the response rowset object from the buffer */
   &response_RS = &response_MSG.GetRowset();

   /* Loop through the message rows moving the data into the database table */
   For &I = 1 To (&response_RS.RowCount)
      &response_REC = &response_RS.GetRow(&I).GetRecord(Record.PSINST_VER_TRX);
      &response_REC.CopyFieldsTo(&IB_INST_VER_DB_REC);
      &IB_INST_VER_DB_REC.Insert();
   End-For;
End-If;

/* Manual refresh of scrollable area */
&IB_INST_VER_TRX_RS = GetLevel0()(1).GetRowset(Scroll.PSINST_VER_TRX);
&IB_INST_VER_TRX_RS.Flush();
&IB_INST_VER_TRX_RS.Select(Record.PSINST_VER_TRX);
```

Glossary

adapter

Provides universal connectivity by enabling an electronic interface to be accommodated (without loss of function) to another electronic interface.

agent

Supports service protocols in listeners and documents.

business service

Also known as a Web service. A Web service is a self-contained, modularized function that can be published and accessed across a network using open standards. It is the implementation of an interface by a component and is an executable entity.

channel

Represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by an adapter.

listener

A component that accepts requests from client applications.

port

Associates a particular business object exposed by the adapter with a particular disposition. A disposition is a URL that defines the protocol and location of the event data. The port defines the end point of the event consumption.

Index